MAYO 1987

Revista del usuario de Drean Commodore

AÑO 2 Nº 17 ₳ 3,40 REP. ARGENTINA



JUAN CARIAGA
MINISTRO DE FINANZAS
DE BOLIVIA

SUMARIO

NOTICIAS

NOTAS TECNICAS

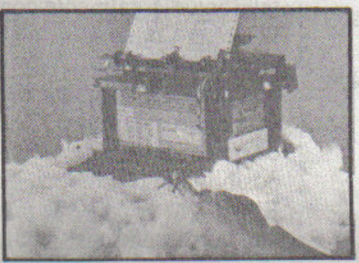
PROGRAMAS

REVISION DE SOFT

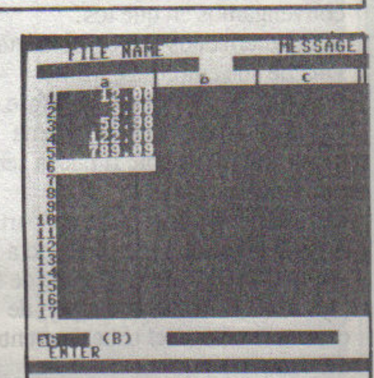
SECCIONES FIJAS



Una Commodore 64, con el agregado de un simple datasete y un televisor color casero, fueron las herramientas utilizadas por Juan Cariaga, Ministro de Finanzas de Bolivia, para sacar a ese país de la hiperinflación.



¿Qué hacen los nuevos sustitutos de las máquinas de escribir? Todos, a través de órdenes como "erase", "insert", "move" o "cut"-"paste", contribuyen a una mayor productividad. Ahora les presentamos sus funciones y mecanismos.



La planilla electrónica, también conocida con el nombre de hoja electrónica, es un programa utilitario que nos permite manipular datos y ecuaciones numéricas cómodamente.



AÑO 2 Nº 17 1987

Director General
Ernesto del Castillo

Director Editorial
Cristián Pusso

Director Periodístico
Fernando Flores

Director Financiero
Javier Campos Malbrán

Secretario de Redacción
Ariel Testori

Redacción
Cristián Parodi

Arte y Diagramación
Fernando Amengual
Tamara Migelson

Fotografía
Victor Grubicy Image Bank
Eduardo Comesaña

Departamento de Avisos
Oscar Devoto Nelso Capello

Departamento de Publicidad
Guillermo González Aldalur

Revista para usuarios de Drean Commodore es una publicación mensual editada por editorial **PROEDI S.A.**, Paraná 720, 5º Pis. (1017) Buenos Aires. Tel.: 46-2886 y 49-7130. Reg. Nac. de la Prop. Intelectual E.T., M. Registrada. Precio de este ejemplar, ₳ 3,40. Los ejemplares atrasados se venderán al precio del último número en circulación. Queda hecho el depósito que indica la Ley 11.723 de Propiedad Intelectual ISSNO326-8233

Impresión: Calcotam. **Fotocromo tapa**: Columbia. **Fotocomposición**: Interamericana Gráfica S.R.L.

Distribuido en Capital: Martino, Juan de Garay 358, P.B. Capital. **Distribuidor interior**: DGP, Hipólito Yrigoyen 1450, Capital Federal. T.E.: 38-9266/9800.

Apple y Commodore 64



Si de compatibilidades se trata, convengamos en que los norteamericanos son especialistas. Una de sus empresas, Minic System, desarrolló una interfase que permite que cualquier programa escrito para la Apple II pueda ser ejecutado en la 64/C.
Esta interfase, denominada Spartan, es un dispositivo que se conecta detrás de la 64/C en cada una de sus puertas de entrada/salida (port de expansión, port del usuario y entrada del datasete).
Su carcasa es similar a la de un disco rígido. En su interior se encuentran 64 Kbytes de RAM para emular una Apple II, el sistema operativo de disco que permite que la 1541 pueda trabajar en modo Commodore y modo Apple y los conectores necesarios que posibilitan el enlace con la consola.
Internamente, podemos encontrar dentro de la interfase cinco slots de expansión que posibilitan, entre otras cosas, aumentar la memoria a 256 Kbytes, conectar impresoras Commodore o tipo Centronics, o conectar otro drive Commodore o Apple.
Uno de los inconvenientes que este dispositivo posee es que no podemos usar el port del usuario, necesario para telecomunicaciones por modem telefónico.
Por otra parte, Spartan permite correr el 100% de los programas comerciales y de aplicación para la Apple II.
Sin embargo, esta compatibilidad no es tal cuando se corren programas desarrollados para Commodore en modo 64/C.
Esto se debe a que Spartan utiliza un área de la ROM de la 64/C que, en algunos casos, es usada por ciertos programas comerciales. Aún no es posible adquirir este dispositivo en nuestro país.

Nuevo modem



PyM-Soft lanzó recientemente al mercado nacional su nuevo modem telefónico, totalmente compatible con la Drean Commodore 64/C y Commodore 128.
Sus dimensiones reducidas (20,5 centímetros de frente, 12 centímetros de profundidad y 5 centímetros de alto) posibilitan una cómoda operación.
Está diseñado para que pueda trabajar con las dos normas actuales de comunicación (Bell y CCITT), con las que accedemos a cualquier banco de datos nacional o extranjero.
Otra de sus características es que su velocidad de transmisión es de 300 baudios. El fabricante asegura que en breve lanzarán el segundo modem que podrá transmitir a 300, 600 y 1200 baudios y, además, tendrá la posibilidad de discado automático (autocall) y respuesta automática (autoanswer).
La alimentación del modem la suministra directamente la computadora. Y su precio al público es de 178 australes existiendo una oferta especial para aquellos que se quieran subscribir a DELPHI.

Epromaster 64&128

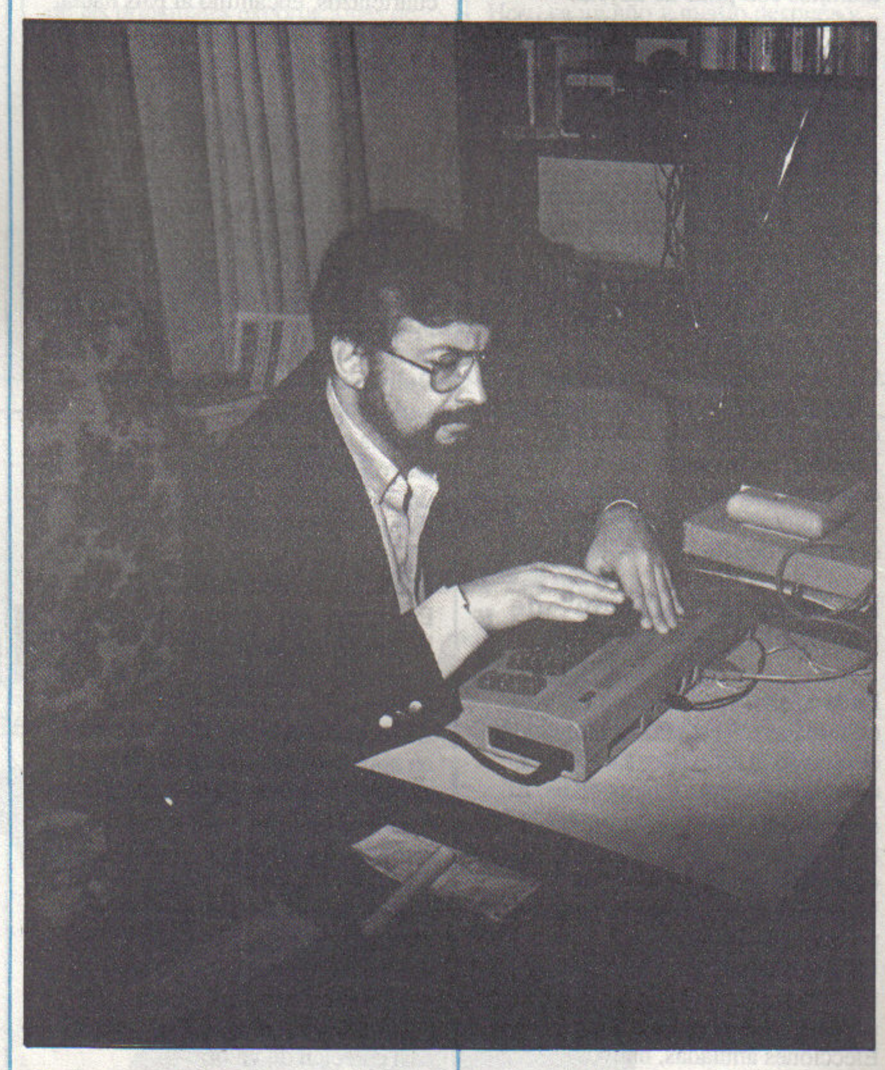
Este dispositivo, controlado por una Drean Commodore 64/C o C-128, permite grabar varios tipos de memorias Eprom.
Como una aplicación inmediata podemos citar la de crear nuestros propios cartridges, los cuales pueden re-programarse.
Recordemos que una de las principales características de las memorias Eproms reside en la posibilidad de borrar la información almacenada con sólo exponerla a la luz ultravioleta.
EPROMASTER 64&128 dispone, dentro de su gabinete, de una pequeña lámpara ultravioleta que, junto con la posibilidad de grabar varios tipos de Eproms, lo convierte en uno de los multi-programadores más potentes del mercado.
Waisman, firma encargada de la elaboración de EPROMASTER 64&128, suministra junto con el dipositivo, un completo manual y una plaqueta especialmente diseñada para la Drean Commodore 64/C y C-128, que contiene una memoria 2764 en la que está grabado el popular juego Frogger.



LA COMMODORE QUE SALVO UN PAIS

Una Commodore 64, con el agregado de un simple datasete y un televisor color casero, fueron las herramientas utilizadas por el joven ministro de Finanzas boliviano, Juan Cariaga, para sacar a ese país de la hiperinflación que amenaza desembocar en un colapso total. El funcionario recibió a Drean-Commodore en su residencia paceña y explicó en una larga entrevista de qué manera su computadora hogareña fue pieza clave en el programa que permitió bajar la inflación del 30 mil por ciento anual a sólo un 60 por ciento.

En agosto de 1985 Bolivia se encaminaba hacia su virtual desaparición como Estado, en una crisis sin precedentes con una inflación superior al 30 mil por ciento anual. Apenas un mes después un paquete económico de emergencia lograba frenar "en seco" la suba de precios reduciendo las tasas a sólo 60 por ciento anual, un verdadero "milagro" que fue ejecutado sobre una computadora hogareña Commodore 64.
El autor de esta mutación fue el joven economista y actual ministro de Finanzas boliviano Juan Cariaga, quién en su casona neocolonial de La Paz permaneció encerrado durante 20 días elaborando en absoluto secreto sobre la C-64 de su hijo los análisis de flujos financieros y los textos del decreto supremo 21.060 que sacaron a Bolivia de la hiperinflación.
Cariaga recibió a **Drean-Commodore** en su residencia del barrio de Calacoto, en La Paz, y en el curso de una prolongada entrevista explicó los procesos que permitieron salvar del colapso a ese país de 6 millones de habitantes con la simple ayuda de la C-64, un datasete y un televisor color casero.



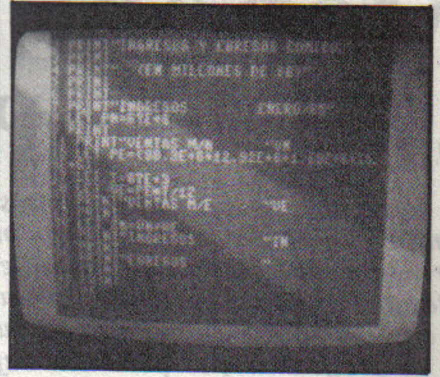
Veinte años después

Toda historia, en realidad, tiene múltiples orígenes, y para éste elegimos 1966, cuando un estudiante de ciencias económicas llega desde Bolivia a Estados Unidos para perfeccionarse.
Juan Cariaga desembarcó hace 21 años en una universidad californiana provisto de su regla de cálculo y dispuesto a adentrarse en los secretos de su disciplina.
"Imagínese mi sorpresa cuando descubrí que esa regla de cálculo que yo llevaba, y que consideraba como el sumum, era ya un anacronismo".
En aquellos años comenzaba la revolución informática, algo que para un estudiante proveniente del Tercer Mundo parecía cosa de ciencia ficción.
En muy pocas clases Cariaga comprendió que debía "tirar a la basura" su regla de cálculo y adentrarse en el estudio de la computación y las matemáticas.
En esta inesperada tarea tuvo la fortuna de contar con un maestro de

primer nivel: nada menos que el profesor Kemeny, el creador del lenguaje BASIC.
Durante tres años Cariaga trabajó con ahínco en sus cursos de ciencias económicas, mientras absorbía los conocimientos de computación y trabajaba sobre los sofisticados -por entonces- equipos General Electric 360.
Cariaga admite que por aquellos años apreciaba con una mezcla de asombro y sorpresa las capacidades de esos gigantescos ordenadores a los cuales alimentaban todos los alumnos con pilas de tarjetas perforadas y, a veces, se preguntaba si esas facilidades llegarían alguna vez a su país y cuando se produciría tal acontecimiento.
"Aún hoy recuerdo una de mis primeras clases con Kemeny, quien con una seguridad que nos sorprendió a todos sus alumnos, nos vaticinó que debíamos esforzarnos por aprender computación porque en 20 años más iba a estar al alcance de todo el mundo."
Cariaga culminó sus estudios en 1969 y en el '71, tras finalizar su carrera de post-grado, volvió a Bolivia, "archivando" la computación durante tres lustros. Aún ignoraba la utilidad que obtendría de ella 15 años después.

La historia "caliente" de un pueblo

En la misma época en que Cariaga culminaba sus estudios y emprendía el regreso al altiplano, La Paz era escenario de una violenta crisis. El gobierno nacionalista de izquierda del general Juan José Torres era desalojado por las armas y la Universidad de La Paz bombardeada por aviones insurrectos.
Un general llamado Hugo Banzer, tomó el poder durante 7 años. La salida de este período fue traumática. Elecciones anuladas, nuevos comicios, denuncias de recuentos fraudulentos, golpes de Estado y un clima de fragilidad política que duró hasta 1982, período durante el cual en cada año se alternaban varios presidentes civiles y otros tantos militares.
Finalmente en el '82 las elecciones consagraron jefe de Estado a Hernán



Siles Suazo, quien si bien logró sortear la costumbre de los cuartelazos, encaminó al país hacia su virtual disgregación por medio de una crisis económica sin precedentes.
Y si para muestra basta un botón, conviene recodar que en noviembre de 1984 en el marco de una de las habituales huelgas generales por tiempo indeterminado de la poderosa Central Obrera Boliviana (COB), aderezada con bloqueos de caminos por sectores campesinos (algo que hizo faltar alimentos en las ciudades) Bolivia conoció el primero de sus varios "Rodrigazos".
En una sola noche un "tarifazo" elevó los precios de los alimentos en un 350 por ciento, los combustibles en niveles que variaron entre 400 y 800 puntos y la cotización del dólar oficial en 90 por ciento.
Era apenas el comienzo. El dólar llegó a 9 mil pesos y pocos meses después ya orillaba los 2 millones de pesos.
La "locura económica" era tal que, en aquellos meses, una de las mayores importaciones de Bolivia era papel moneda, ya que cada semana se lanzaban a la calle billetes de mayor denominación. Adquirir apenas los alimentos para una semana, obligaba a transportar un maletín repleto de papeles de colores que, en otra época, hubiesen sido denominados billetes, pero que ya casi carecían de valor.

La salida

A comienzos de 1985 Cariaga, quien luego de tres lustros dedicados a la actividad privada había ganado un sitio entre los jóvenes economistas bolivianos, viajó a Harward, Estados Unidos, para participar en una conferencia sobre la hiperinflación.
"Una tarde, paseando por la ciudad, en una tienda encontré una Commodore 64 con su datasete a un precio apenas superior a los 200 dólares y en ese momento recordé la predicción del profesor Kemeny 20 años antes."
"Mucho mayor fue mi sorpresa cuando, al estudiar las especificaciones técnicas de la máquina y probarla, descubrí que era capaz de realizar las mismas cosas que a mí me asombraban dos décadas atrás, cuando trabajamos sobre las General Electric 360" añadió el ministro, quien de inmediato la compró para regalársela a su hijo, aún en la escuela primaria.
Pocas semanas después se realizaron elecciones anticipadas en Bolivia y ninguno de los tres candidatos principales logró la mayoría de la mitad más uno de los sufragios.
Conforme a la Constitución, fue el Parlamento, entonces, el que eligió al nuevo presidente, el anciano Víctor Paz Estenssoro, de 80 años, y líder del Movimiento Nacionalista Revolucionario Histórico.
"El doctor Paz me ofreció el ministerio de Finanzas, pero por aquella época yo tenía una serie de compromisos que me impedían aceptar. Entonces me pidió que fuese su asesor, a lo que accedí gustoso", rememoró Cariaga.
El 6 de agosto Paz juró como presidente y al día siguiente convocó a Cariaga de urgencia a su despacho.
El ahora ministro recuerda así aquella entrevista: "Paz nos dijo que el país se encaminaba hacia el colapso económico total en el corto plazo y por eso había designado un grupo de seis especialistas, entre los cuales me contaba yo, para que en un plazo de 20 días, y en absoluto secreto, elaboráramos un programa que salvara a Bolivia".
Las condiciones, según recordó Cariaga, eran que fuese un plan simple y de efectos rápidos, capaz de evitar el colapso -y eventual "desplome" del Estado-, fácil de aplicar y que no se filtrara ninguno de sus elementos (ni siquiera debía conocerse su proyecto) para evitar la especulación adicional que ello hubiese provocado.
"El requisito del secreto absoluto

APLICACIONES INEDITAS

nos impedía trabajar en cualquier oficina gubernamental, así que entonces me ví forzado a recluirme en el estudio de mi casa y a partir de allí comenzar todos los trabajos".
Cariaga, como primer paso, practicó una suerte de "expropiación transitoria" de la Commodore 64 de su hijo y, munido de sus viejos libros de Basic, empleó siete días completos en preparar un programa de análisis de flujos financieros.
"El problema -recuerda ahora con humor- es que yo había estado alejado de la computación durante muchos años e ignoraba que ese mismo programa que yo estaba diseñando desde cero era el Lotus, un utilitario que desde hace tiempo está a la venta".
El objetivo fundamental era cortar de raíz el gasto público para no emitir moneda sin respaldo y Cariaga trabajó durante días efectuando los análisis de flujos financieros en las principales empresas estatales para cotejar los ingresos reales y cuánto era lo que podían gastar, o por dónde había que recortar las salidas.
"Fue algo agotador. Ni siquiera tenía una impresora, sólo el teclado, la grabadora de casete y el televisor, así que cuando alcanzaba un resultado, tenía que copiar todo manuscrito, porque si hubiese tratado de gestionar una impresora, alguien podría haber sospechado algo", relató el funcionario en el estudio de su casa.
"Muchas tardes venía hasta aquí el Presidente Paz y en ese mismo sillón en el que usted está sentado me preguntaba qué pasaría por ejemplo si el precio del estaño (una de las principales exportaciones de Bolivia y eje de la minería estatal) bajara, por ejemplo, otro 3 por ciento.
Cariaga en su relato elogia la paciencia del octogenario presidente ya que "debía aguardar muchos minutos hasta que con el datasete encontrara el programa correspondiente a la COMIBOL (Corporación Mineral Boliviana) y una vez alimentada la Commodore con los nuevos datos, le copiara manuscrito desde el televisor los resultados".
En tres semanas de trabajo el equipo de dos economistas, otros tantos empresarios y dos políticos culminó el plan, bajo la conducción de Cariaga.
Su anuncio, bajo la forma del decreto supremo 21.060, movió una "ola de confianza" en todo el país que, conjugada con la total disciplina fiscal de limitar los gastos a los ingresos reales (control efectuado a través de la C-64 de Cariaga) salvó a Bolivia del colapso.
Hoy, ya un año y medio después y sin los impedimentos del secreto absoluto de aquellos 20 días, los controles de gestión se efectuán a través de un equipo de computadoras IBM instaladas en el ministerio de Finanzas, a cuya titularidad accedió Cariaga a fines del año '85.
En cuanto a la Commodore 64 que sirvió para diseñar el plan que salvó a Bolivia, fue restituía al hijo del ministro.

Marcelo Brusa

EL USUARIO DEL AÑO

LA DREAN COMMODORE MAS POLAR DEL MUNDO

Conversamos con el responsable de la instalación de una Drean Commodore 64 en la Antártida Argentina, que sirvió para comunicarse con todo el mundo.

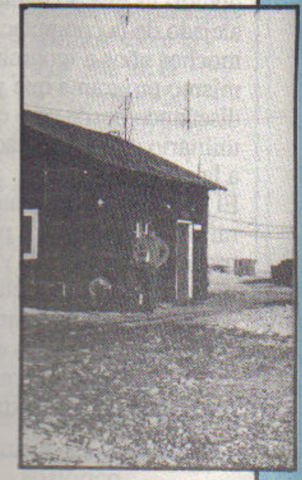
Juan Carlos Parra fue el ganador de la tercera mención de nuestro concurso "El usuario del año", por un proyecto que él mismo dirigió e implementó.
Relacionado con las comunicaciones, Parra creó una interfase que le permite comunicar su Drean Commodore 64 a un equipo tranceptor, es decir aquél que usan los radioaficionados para comunicarse con otros de su tipo.
De esta manera, y gracias a un programa que también diseñó (llamado SYS 2000), utiliza su Drean Commodore 64 para ingresar los mensajes que luego se enviarán al "aire".

El proyecto, sin embargo, no termina ahí. Parra, quien es Ingeniero Electrónico, trabajando para la Armada Argentina con el grado de teniente de navío ingeniero, tuvo que realizar tareas de mantenimiento en una base antártica.
Aprovechando la oportunidad, llevó todo su equipo y la interfase que él diseñó. A través de los tranceptores de la base y usando un identificatorio otorgado especialmente por la Secretaria de Comunicaciones, inauguró la estación de radioaficionados más austral del mundo.

Durante los dos meses en que la estación estuvo transmitiendo, Parra se comunicó con más de 2500 estaciones ubicadas en todo el mundo.
Verificó, además, que la computadora trabaja perfectamente en condiciones no ideales de temperatura y humedad.
Por otra parte, y ante nuestra pregunta de por que eligió la Antártida, nos contestó que "para hacer, aún más, soberanía".

Juan Carlo Parra con su equipo instalado en la base antártica.





Desde los 14 años

Por supuesto, alguien que desarrolla un proyecto como el que describimos es porque hace mucho tiempo que está en el tema.

Desde los 14 años, Parra fue un entusiasta radioaficionado que construyó tranceptores del tipo "casero".

En el '83, y gracias a un viaje que realizó a Holanda, descubrió la C64, computadora que compró exclusivamente por todo lo que podía hacer.
"En ese entonces la pagué cara, aproximadamente 500 dólares, no debemos olvidar que recién aparecía en Europa", nos comentó.
"Cuando regresé al país me encontré con que la TI 99 tenía todo acaparado. Muchos colegas me dijeron que la Commodore no iba andar", agregó.
Con el pasar del tiempo vio que aquellos comentarios del "no va andar" se fueron perdiendo en la oscuridad. Poco a poco la Commodore iba creciendo y se afianzaba más y más en el mercado nacional.

En su casa, ubicada en el barrio de Olivos, Parra dispone de una estación de radioaficionado constituida por una antena direccionable y cuatro tranceptores profesionales.
"Este es mi centro de operaciones, es mi cueva", nos dijo.
Fue entonces cuando demostró todas las posibilidades de su invento. Para ello, prendió uno de sus equipos sintonizándolo en frecuncia de radioteletipo.
El mensaje que iba a transmitir lo tipeó en la Commodore 64 para que luego, y con sólo presionar un botón, fuese enviado al aire.
El programa, además, permite hacer otras cosas, como tener la pantalla divida en dos, viendo en una mitad lo que se transmite y en la otra lo que se recibe. También puede mostrar continuamente la hora, y tiene disponibilidad de 8 memorias en las cuales ingresa mensajes de hasta 500 caracteres, y el buffer de transmisión/recepción. Son sólo algunas de las posibilidades de SYS 2000.
Como verán, el concurso del "Usuario del año" resaltó los mejores trabajos.



El hábito hace al lector.

Al lector de revistas se lo define por sus hábitos.
El 41% de ellos es fiel. Ha encontrado la revista que lo satisface y no piensa cambiarla por otra. La compra regularmente y sabe disfrutarla con intensidad.
El 49% busca entre las distintas revistas para hallar finalmente la de su agrado.
Entre estos dos estilos de lector de revistas, ¿cuál es el suyo?

Datos resultantes de la encuesta realizada por la Asociación Argentina de Editores de Revistas sobre Hábitos de Compra, con el fin de determinar la habitualidad de compra de los lectores.

Las revistas y usted.

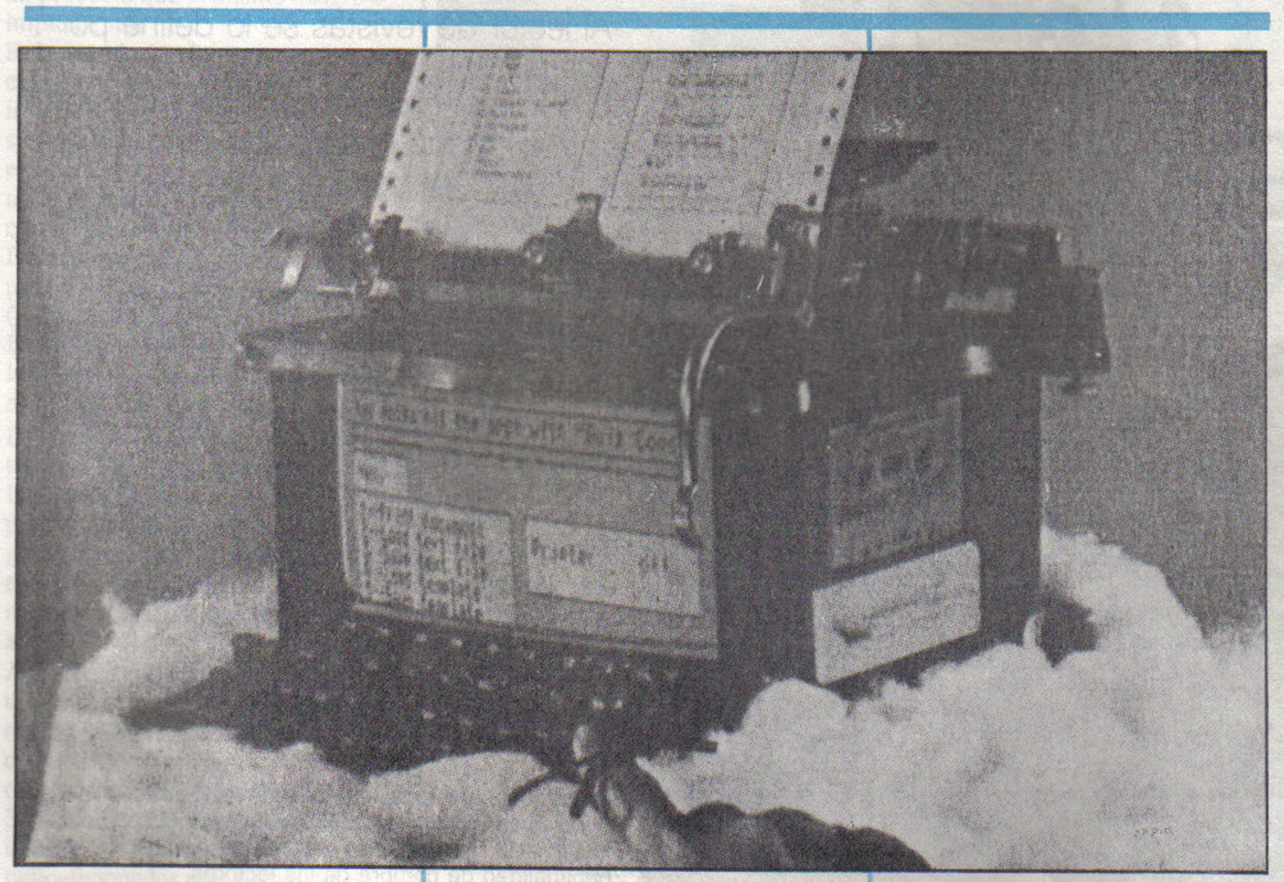
Una relación sin límites.



LOS PROCESADORES DE TEXTO

(1º PARTE)

¿Qué hacen los nuevos sustitutos de las máquinas de escribir?
Todos, a través de órdenes como "erase", "insert", "move" o "cut"-"paste", contribuyen a una mayor productividad. Ahora, les presentamos sus funciones y mecanismo.



Hasta su irrupción, hará aproximadamente quince años, corregir era sinómimo de reescribir . Y no sólo la parte alterada . La elaboración de un borrador, cuando no de sucesivos borradores, resultaba ineludible incluso hasta para el envío de la más anodina carta comercial .Mejor ni pensar en escritos de mayor envergadura o con otro tipo de aspiraciones .
Sin duda, de hecho, estamos frente a una silenciosa revolución. Ha llegado a tal punto la falta de estridencia que ni siquiera se conoce el nombre del primer diseñador o por lo menos el del que tuvo la idea. La magnitud del acontecimiento lo puede dar un hecho como que con la creciente incorporación de los procesadores de texto se terminó el reinado de la máquina de escribir, un monopolio que se extendió durante casi un siglo, y que a partir de las bolitas, las teclas correctoras, margaritas y visores de cuarzo amenazaba con perpetuarse otro poco. Además, las cualidades y ventajas que ofrecen ha determinado que hoy sean uno de los principales motivos para la venta masiva de computadoras hogareñas, con la secuela cultural que resulta de esto para la vulgarización de la informática.
Desde el punto de vista de facilidades y logros, la alternativa máquina de escribir versus procesador de texto no existe. Lo que sí puede llegar a convertirse en un dilema difícil de superar es cuál de ellos es, si no el mejor, por lo menos el más

conveniente. El increíble desarrollo que ha tenido hace que actualmente haya para cubrir prácticamente todas las necesidades. A veces son detalles o pequeñas extravagancias lo que diferencia a uno de otro.

Definir y Precisar

Antes de abordar algún tipo de definición, resulta mucho más útil ver lo que hace. Las funciones básicas son sólo tres. De allí se derivan todas las demás. El agregado entre paréntesis indica las denominaciones en inglés más usadas :

* borrar (**erase/delete**)
* insertar (**insert/add**)
* mover (**move/paste**).

Borrar parte de un texto, por ejemplo, no constituiría una novedad demasiado importante, máxime si se tiene en cuenta la variedad de gomas que hay circundando, los líquidos, cintas y lápices correctores, esto sin dejar de lado la etapa histórica cumplida por la equis en la vieja máquina de escribir, a tal punto que cuando llegó la electrificación se le puso a la tecla una función repetidora igual que al subrayado mayúscula del 6. Sin embargo, todo este arsenal de despliege tecnológico jamás había logrado algo que se creía insuperable y a lo que nos habíamos resignado : el espacio raspado, pintado o lleno de tachaduras que cada enmienda inevitablemente traía como consecuencia. En la cabeza de nadie podía caber que de un texto se volara una, dos, tres, o cuarenta y siete palabras y no quedara un solo rastro de semejante procedimiento. No sólo esto. Porque lo de no dejar ni un rastro no debe ser tomado tan al pie de la letra. Según el procesador de que se trate, lo eliminado puede ser o no rescatado. De ahí el distingo que suelen hacer entre borrar (**erase**) y tachar (**delete**). Aquellos procesadores que permiten la recuperación envían la tachadura a un espacio de memoria buffer que es el mismo que se utiliza para las operaciones de mover (**move**) o pegotear (**paste**) , que veremos enseguida. Sin ninguna duda, los que tienen esta capacidad ahorran un sinfín de dolores de cabeza ante apresuramientos, errores o cambios de criterio. Algunos de ellos, como es el caso de **The Bank Street Writer**, la tienen en forma explícita: **erase/unerase** (borrar/recuperar) y **move/unmove** (mover/volver a su lugar) .
La función de insertar texto es exactamente la contraria de borrar y ostenta la misma característica : salvo alguna discordancia de género o número, errores de sintaxis o de concepto, no queda ningún rastro . El resto del texto, el que está a la derecha de la inserción, es reacomodado correlativamente y sin más límite que la capacidad del procesador . O sea que si se está elaborando un escrito y recién al final se ha omitido una palabra, una cita, un párrafo o varias páginas, el único trabajo es volver con el cursor hasta el lugar exacto, activar la función y realizar el injerto.
Por último, la función de mover o pegotear quizá sea la que más se ajuste a la noción exacta de procesar un texto. Facilitar de tal modo que sólo dependa de nuestra voluntad cambiar totalmente el orden de palabras, oraciones o párrafos enteros, ver cómo quedan, volverlos al lugar anterior o todavía producir cuantas alteraciones vengan en gana es algo que permite apreciar y evaluar las diferentes posibilidades, lecturas o simples alteraciones sin entramparse en ese escollo real que es la reescritura.
Para terminar con las características más tangibles que diferencian a los procesadores, así como ninguno que se precie de tal puede no borrar, carecer de la facutad de insertar o de cambiar partes del texto de lugar, la manera en que los caracteres son ordenados en la pantalla sí establece dos grandes categorías. Todos sin excepción eliminan para siempre la preocupación del margen derecho, como en la antigua máquina de escribir, donde había que cortar la palabra al final de la sílaba y, con la palanca o la tecla correspondiente, volver al carro. Los procesadores de texto, salvo para indicar el punto y aparte, señal del inicio de un nuevo párrafo, no necesitan orden en especial,pero al llegar al límite máximo de la línea están los que cortan la palabra completa (**wordwrap**) , mandándola al comienzo de la línea siguiente, y los de flujo continuo (**streaming**) . Simplemente por un problema de costumbre, de falta de hábito, los primeros son los que resultan más simpáticos. Cuesta acostumbrarse a que en pantalla la palabra **subrepticio** quede casualmente dividida en **sub** y **repticio** o **subreptic** y a renglón seguido **io**. Es lo único, porque por impresora todos sacan el texto con el espaciado correspondiente, incluso justificando el margen derecho, si es que se lo ordena.

Otras Capacidades

Sobre las tres funciones básicas descriptas recién giran muchas más. Cortar y pegar, por ejemplo, función que da que nombre a un buen procesador como es **Cut & Paste**, no es otra cosa que la parte de un texto que ha sido tachada vaya a parar a la buffer y que luego sea rescatada y "pegada" en la nueva posición que indica el cursor. La acción de cortar (**cut**) sólo tiene con la de tachar (**delete**) una diferencia de nomenclatura, ya que en los hechos son idénticas.
Lo mismo ocurre con la de repetir (**repeat** o **copy**, según el procesador), sumamente útil en textos comerciales o publicitarios, como así también en aquellos en que por cualquier motivo una palabra, oración o párrafo deben aparecer muchas veces. En realidad, el texto marcado para ser repetido o copiado lo que sufre es una "tachadadura", va a parar a la buffer y de allí se lo saca para ser pegado o movido cuantas veces se lo desee y en el lugar deseado, con sólo poner el cursor en el lugar correspondiente y dar la orden respectiva.
Ahora bien, en lo que hace al pulimiento de un texto, la función de buscar (**find search** o **hunt**, según el que se use) , por lo común indisolublemente unida a la de reemplazar (**replace**) , aparece como una de las características más distintivas de este indispensable utilitario. La manera en que funciona el mecanismo de buscar/reemplazar depende de cada procesador, pero las diferencias hacen a matices secundarios. Absolutamente todos permiten encontrar determinada

palabra, conjunto de palabras,oraciones y a veces hasta párrafos muy cortos a través de todo un texto, cualquiera sea la extensión de éste. Algunos, por ejemplo, ofrecen el reemplazo directo y automático, a otros hay que ordenárselo específicamente cada vez, pero se trata de capacidades que nunca están ausentes.

No sólo más, sino mejor

Por supuesto, la facultad made in USA que más se ha publicitado de los procesadores es su contribución a la productividad. Algo innegable : se puede escribir mucho más en menos tiempo. Ahora si esto por sí mismo constituye un mérito o algo atrayente, dejémoslo así. Pero no por eso hay que dejar de lado que el atributo principal está en las facilidades que otorga incluso para que esa mayor cantidad sea mejor, una calidad que va desde la posibilidad de corregir los errores de tipeo a la claridad o elaboración de un escrito.
La simpleza y ventajas que tiene son mucho más trabajosas de explicar que de experimentar. Por lo tanto, tomemos un ejemplo muy simple. Supongamos que para elaborar un texto sobre maquinaria agricola hay que andar tipeando estas dos palabras a cada línea y media. Como el "procesador de ideas" que es el cerebro camina a velocidades mucho más fantásticas que los torpes dedos del mejor de los dactilógrafos, para ahorrar tiempo y ganarlo en aguzar la unión entre cerebro y palabras, lo que hago es poner simplemente **ma** o **xx** cada vez, y una vez terminado, buscar la opción **search**, **find** o **hunt**, tipear lo que usé y, cuando en la línea de comandos se me pregunta **Replace with** ?, simplemente escribo **maquinaria agrícola**, ingreso la orden con <RETURN> y me quedo a mirar cómo trabaja por mí o aprovecho para estirar el cuerpo, hacer una llamada o calentar el café.
Digamos que esta es una aplicación posible, la más directa y banal. Pero cuando se trata de una elaboración más depurada de un texto, la función de búsqueda-reemplazo llega a cumplir tareas de un valor incalculable. Porque así se trate de una persona que no escriba profesionalmente, quien más, quien menos, conoce sus falencias estilísticas, tics, tendencias a ciertos lugares comunes, construcciones clishé, redundancia, o el peligro, como en el caso antes apuntado, de que "maquinaria agrícola" esté repetido tantas veces que en vez de aclarar se convierta en ripio para el lector. Un chequeo mediante esta opción permite apreciar este tipo de falencias y, de paso, ir haciendo las correcciones que se crean necesarias. También hay que recurrir a ella cuando el procesador tiene la capacidad de hacer recuento estadístico de cantidad de palabras, tanto del total como de las que están usadas más de una vez y cuántas veces cada una . Tal es el caso de **Better Working**, que precisamente permite tener en cualquier instante un panorama de este tipo.

Paso Final

Al permitir todo tipo de facilidades y licencias para elaborar un texto sin tener que recurrir a la impresión en ninguna de sus variantes, como no sea al final, cuando uno ya ha decidido que está todo correcto, empezando por el tipeo, lo que ofrece como realmente fascinante y maravilloso el procesador es poner en un hogar alternativas que hasta ahora eran privativas sólo de imprentas comerciales y tirajes considerables. Insospechada variedad de tipografías y tamaños, poder recurrir a la negrita o la bastardilla para resaltar partes, centralizar títulos y subtítulos, justificar también sobre margen derecho, poder cambiar el interlineado y el **paso** (espacio) entre una letra y otra, donde jamás la "i" y la "m" pueden ocupar lo mismo, ahora pueden incorporarse como elementos indispensables y constitutivos del escrito que se ha elaborado. Otro tanto ocurre con la numeración automática y correlativa de las páginas, la posibilidad de ponerles leyendas de cabezal o al pie y cantidad de comandos de impresión más.
Cuando se aborda este aspecto, surge inevitablemente otro : contar con un procesador y hacer uso de él, por lógica consecuencia, indicaría que hay que contar con una impresora. La repuesta obvia es que sí, pero no es una condición sine qua non. Primero porque la impresora no tiene por qué ser de uso oficial y exclusivo; segundo porque todavía la difusión de los procesadores y la crisis económica no han hecho que así como nos fotoduplican una hoja en cuestión de horas, esté cercano el día que las publicaciones especializadas tengan avisos donde se nos ofrezca ese tipo de servicios. Tercero porque determinados textos, sobre todo si son muy personales o exigen una larga elaboración, pueden ser tranquilamente guardados en disquete, donde ocupan mucho menos lugar. Por último, en cuarto lugar, porque la cada día mayor divulgación de la informática va a hacer que cada vez menos trabajos escritos sean entregados sobre papel. En lo personal, no hace mucho a un trabajo por encargo, que seguramente llegará a ocupar más de 80 carillas comunes, lo llevé en un bolsillo de la campera : eran sólo 390 bloques en una cara del disquete. El destinatario sabrá qué formato darle, qué tipo de letra y demás características.
No es todo lo que hay y se debe decir. Pero tampoco vayan a creer que queda mucho por agregar. Sin embargo, pensando en que pueda haber alguien que todavía no le vea mucho asunto o demasiadas diferencias a favor frente a una máquina de escribir, lo único que se le puede responder es que pruebe. Como adelanto, desde ya le anticipamos que la diferencia es mucho mayor que la de usar papel o un tubo de rayos catódicos como superficie. También que si se opta por lo tradicional, no se atormente ni se haga malasangre. A pesar de que todavía no existía el periodismo, han quedado cantidad de trascendidos sobre gente que en su época desechó la rueda por considerarla un invento estrafalario o que pusieron el grito en el cielo por el peligro que entrañaba el fuego. No dejaban de tener sus razones; sólo que no hubo forma de detenerla o apagarlo.

AMILCAR ROMERO

Continuará

GENERADOR DE CALENDARIOS

ABRIL 1987						
DOM	LUN	MAR	MIE	JUE	VIE	SAB
		01	02	03	04	05
06	07	08	09	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

Comp.: Drean Commodore 64/C
Conf.: Consola y/o impresora
Tipo: Utilitario

Este programa nos permitirá imprimir por impresora el mes de cualquier año comprendido entre 1800 y 2000.
Al ejecutarlo el programa nos pedirá que ingresemos el número del mes y el año que se imprimirá.
También nos indica cuál debe ser el formato de los datos a ingresar (MM/AAAA). Por ejemplo, si queremos emitir el mes de abril de 1987, debemos responder a la pregunta con: 04,1987.
A partir de aquí se comenzará a confeccionar por impresora el mes solicitado.
En lo que respecta al tipo de impresora, el programa puede trabajar con cualquier tipo de la serie MPS de Commodore (801, 802, 803, 1000 y 1200).
Los que tengan impresora deberán tipear el listado 1, mientras aquellos que no la tengan tendrán que tipear el listado 2.
Este último sacará por pantalla el mes pedido.

Estructura del programa

Línea	Descripción
5	Setea color de fondo, borde y del cursor
10	Dimensiona los vectores que se utilizarán
25-30	Lee los meses y los días guardándolos en el vector y en un string
45	Pide el mes y el año a imprimir
100-110	Datas conteniendo meses y días
120-125	Cálculo del año bisiesto
200-240	Subrutina: imprime encabezamientos y lee el teclado
300-390	Imprime título del mes y los días

Variables utilizadas

Nombre	Descripción
T$	Separador horizontal. Contiene 40 signos menos
C$	Separador vertical. Contiene el caracter SHIF H
NO,MO	Vectores: indican días por mes y nombre del mes
J,Q	Variables utilizadas en cálculos diversos
CE	Centrado del mes y del año
LO,L1,L2,L3	Variables índices usadas en lazos FOR-NEXT

Listado 1

```
1 REM ****************************
2 REM * GENERADOR DE CALENDARIOS *
3 REM *                          *
4 REM ****************************
5 POKE53280,0:POKE53281,0:PRINT"▒"
6 REM SE INICIALIZAN LOS VECTORES QUE          CONTENDRAN LOS MESES Y DIAS
7 :
8 :
9 :
10 DIM ND(12),MO$(12):OPEN4,4
11 :
12 :
13 :
14 :
15 T$="
20 C$="|            ":T$="        "+T$
25 FORLO=1TO12:READND(LO),MO$(LO):NEXTLO
30 FORLL=1TO7:READDY$:AA$=AA$+"        "+DY$:NEXT
40 GOSUB200
41 :
42 :
43 :
44 :
45 INPUT"INGRESE MES Y A#O (MM,AAAA)";MO,YR
46 :
47 :
48 :
49 :
50 GOSUB120
55 XX=(ND(MO)+J)/7:NU=INT(XX):IFXX<>NUTHENNU=NU+1
60 IF (YR/4)-INT(YR/4)=0 THEN ND(2)=29
65 SR=(J*10)+4
70 GOSUB300
75 CLR:INPUT"OTRO MAS (S/N)";XZ$:IFXZ$="S"THENCLOSE4:RUN
80 CLOSE4
85 END
90 GOSUB 300
95 END
96 :
97 :
98 :
```

```
90 REM DATAS DE LOS MESES Y DIAS
100 DATA 31,ENERO,28,ENERO,31,MARZO,30,ABRIL,31,MAYO,30,JUNIO,31,JULIO,31
105 DATA AGOSTO,30,SEPTIEMBRE,31,OCTUBRE,30,NOVIEMBRE,31,DICIEMBRE
110 DATA DOM,LUN,MAR,MIE,JUE,VIE,SAB
111 :
112 :
113 :
120 Q=YR+(MO(3):J=INT(275*MO/9)-INT((7*Q+7)/4)+367*YR+D-INT((INT(Q/100)+1)*3/4)
125 J=J-4-7*INT((J-4)/7):RETURN
130 RETURN
131 :
132 :
133 :
134 :
135 DATA DOM,LUN,MAR,MIE,JUE,VIE,SAB
136 :
137 :
138 :
200 PRINT"J":PRINT"GENERADOR DE CALENDARIOS"
210 PRINT
215 PRINT
220 PRINT
225 PRINT
230 PRINT "PRESIONE UNA TECLA CUANDO LA IMPRESORA  ESTE LISTA"
235 GETG$:IFG$=""THEN235
240 RETURN
300 CE=(64-LEN(MO$(MO))*2-12)/2+3:PRINT#4:PRINT#4,SPC(CE)CHR$(14);MO$(MO)+" ";Y
305 PRINT#4,CHR$(15):PRINT#4
310 IN=4:GU=0:N3$="01":N3=1
315 PRINT#4,SPC(7)AA$
320 PRINT#4,T$
325 FORLO=1TONU
330 PRINT#4,"          ";
335 FORL3=1TO7
340 IN=IN+3:IFIN>=SRTHENGU=1
345 IFN3>ND(MO)THENGU=0
350 IFGU=0THEN360
355 PRINT#4,LEFT$(C$,7);RIGHT$(N3$,2);:N3=N3+1:N3$="0"+MID$(STR$(N3),2):GOTO365
360 PRINT#4,C$;
365 NEXTL3:PRINT#4," |"
370 FORL2=2TO8
375 PRINT#4,"          ";C$;C$;C$;C$;C$;C$;C$;" |":NEXTL2
380 PRINT#4,T$
385 NEXTLO:PRINT#4,CHR$(12)
390 RETURN
READY.
```

Listado 2

```
5 POKE53280,0:POKE53281,0:PRINT"J"
10 DIM ND(12),MO$(12)
15 T$="------------------------------------------"
20 C$=" |        ";
25 FORLO=1TO12:READND(LO),MO$(LO):NEXTLO
30 FORLL=1TO7:READDY$:AA$=AA$+"       "+DY$:NEXT
40 GOSUB200
45 INPUT"INGRESE MES Y A#O (MM,AAAA)";MO,YR
50 GOSUB120
55 XX=(ND(MO)+J)/7:NU=INT(XX):IFXX>NUTHENNU=NU+1
60 IF (YR/4)-INT(YR/4)=0 THEN ND(2)=29
65 SR=(J*10)+4
70 GOSUB300
75 CLR:INPUT"OTRO MAS (S/N)";XZ$:IFXZ$="S"THENCLOSE4:RUN
80 :
85 END
90 GOSUB 300
95 END
99 REM DIA #
100 DATA 31,ENERO,28,ENERO,31,MARZO,30,ABRIL,31,MAYO,30,JUNIO,31,JULIO,31
105 DATA AGOSTO,30,SEPTIEMBRE,31,OCTUBRE,30,NOVIEMBRE,31,DICIEMBRE
110 DATA DOM,LUN,MAR,MIE,JUE,VIE,SAB
120 Q=YR+(MO<3):J=INT(275*MO/9)-INT((7*Q+7)/4)+367*YR+D-INT((INT(Q/100)+1)*3/4)
125 J=J-4-7*INT((J-4)/7):RETURN
130 RETURN
135 DATADOM,LUN,MAR,MIE,JUE,VIE,SAB
200 PRINT"J":PRINT"GENERADOR DE CALENDARIOS"
210 PRINT
215 PRINT
220 PRINT
225 PRINT
230 PRINT "PRESIONE UNA TECLA CUANDO ESTE LISTO"
235 GETG$:IFG$=""THEN235
240 RETURN
300 CE=(30-LEN(MO$(MO))*2-12)/2+3:PRINT:PRINTSPC(CE);MO$(MO)+" ";YR
305 PRINT:PRINT
310 IN=4:GU=0:N3$="01":N3=1
315 PRINTSPC(7)AA$
320 PRINTT$
325 FORLO=1TONU
330 PRINT"          ";
335 FORL3=1TO7
340 IN=IN+3:IFIN>=SRTHENGU=1
345 IFN3>ND(MO)THENGU=0
350 IFGU=0THEN360
355 PRINTLEFT$(C$,7);RIGHT$(N3$,2);:N3=N3+1:N3$="0"+MID$(STR$(N3),2):GOTO365
360 PRINTC$;
365 NEXTL3:PRINT" |"
370 FORL2=2TO2
375 PRINTC$;C$;C$;" |":NEXTL2
380 PRINTT$
385 NEXTLO:PRINT
390 RETURN
READY.
```

PLANILLA ELECTRONICA SWIFTCALC

La planilla electrónica, también conocida con el nombre de hoja electrónica, es un programa utilitario que nos permite manipular datos y ecuaciones numéricas cómodamente.

Como su nombre lo indica, la estructura básica de este utilitario es la de una planilla.

Está formada por una matriz en donde se representan dos coordenadas: las filas numeradas de uno en uno y las columnas que se representan a través de letras.

La planilla queda entonces formada por un reticulado constituido por celdas que a su vez contendrán números o ecuaciones matemáticas (ver figura 1).

Cada celda puede ser seleccionada indicando la fila y la columna dentro de ese reticulado.

A mayor cantidad de celdas por planilla, mayor será la cantidad de operaciones que podremos realizar. Es por eso que uno de los criterios que se tienen en cuenta a la hora de seleccionar una hoja electrónica es la cantidad de filas y columnas que ella posee.

La planilla del SwiftCalc está formada por 104 columnas referenciadas desde la "a" hasta la "zz" (a,b,...,z,aa,ab,...,az,...,za,zb,zz) y por 250 filas, comenzando con 1 y finalizando con el número 250. Utilizando las teclas de movimiento del cursor podemos posicionarnos en cualquiera de las celdas de la planilla, que dispone de 104x250=26000 celdas, cada una de las cuales puede tener valores numéricos, ecuaciones o títulos.

En el primer caso, y siempre hablando de las posibilidades del SwiftCalc, cada número que ingresemos a la planilla puede tener hasta una cantidad máxima de 40 dígitos.

Sin embargo, y debido al ancho de

```
FILE NAME                MESSAGE
      a          b          c          d
1    12.00                Libro      Precio
2     3.00                C-128        12.32
3    56.98                C-64         43.98
4   122.00
5   789.09      33.00Total:            56.30
6   193.98     323.30
7
8
9 Esto es un ejemplo del uso del SC
10
11
12
```

cada celda, sólo podremos ver hasta el dígito 12.

En caso de ingresar un número que supere el tamaño de la celda, SwiftCalc pondrá un mensaje informándonos del tal hecho.

Si vamos ha trabajar con números muy grandes, SwiftCalc permite que los números se ingresen en notación exponencial.

La figura 2 muestra una carga de datos númericos, de hasta 12 dígitos de largo y 2 lugares para decimales. SwiftCalc justifica a izquierda cada uno de los valores númericos ingresados, de manera tal que todos los números están perfectamente alineados.

Por otro lado, y como ya dijimos, es posible también ingresar ecuaciones, éstas nos permiten relacionar una o todas las celdas del SwiftCalc. Podemos decir que en tal celda vaya el resultado de sumar la celda ubicada en la coordenada a1 y la ubicada en c3.

Además, SwiftCalc suministra una serie de funciones matemáticas como por ejemplo sumatorias, promedios, porcentuales, etcetera, permitiendo que la complejidad de la ecuación sea, prácticamente, ilimitada.

Otro tipo de datos que podemos incluir en nuestra planilla corresponde a literales o a títulos. Por ejemplo, podemos desarrollar una planilla que esté formada como se vé en la figura 3.

Cada uno de esos títulos fueron ingresados como si se tratase de números. Swiftcalc reconoce automáticamente cuando ingresamos caracteres en lugar de números.

Bajo el título de "Lista de Precios", nuestra planilla fue confeccionada de la siguiente manera :

1) Nos posicionamos con el cursor en la coordenada a1 y tipeamos "Artículo".

2) Con las mismas teclas del cursor nos trasladamos a la coordenada b1 para luego tipear "Precio".

3) Debajo de cada item, ingresamos los valores correspondientes.

Debajo del "Artículo 5" ingresamos el item "total" (coordenada a6). En vez de realizar la suma a mano, utilizamos la función SUM del SwiftCalc, que realiza la sumatoria de las coordenadas específicas.

En nuestro caso nos ubicamos en b6 y presionamos la tecla "=".

DREAN COMMODORE 64-C

SwiftCalc nos pedirá que ingresemos la fórmula o ecuación que determinará el valor de esa coordenada.
Con sólo tipear SUM (b2:b5), se reemplazará el contenido de esa celda con la sumatoria de los valores numéricos desde la fila 2 hasta la 5 inclusive.
Además, cada vez que pasemos por b6 podremos ver en la parte inferior de la pantalla cómo se ha logrado ese valor. Esto ocurre para cada una de las celdas de la planilla. SwiftCalc nos dice cómo fue conseguido su valor.
Regresando a nuestro ejemplo, si en otro momento vamos a modificar algún precio, SwiftCalc efectúa nuevamente la sumatoria, actualizando el valor de la coordenada b6.
Otro ejemplo que podemos mencionar es aquél en donde la lista de precios se actualiza con sólo modificar un índice.
Para este caso, cada precio se determina en función de ese índice, multiplicando su valor por él.
Para implementarlo usando SwiftCalc, debemos ingresar en la coordenada c2 el valor del índice.
Luego tipeamos los precios en función de esa coordenada. De esta manera cada precio se logra a través de la multiplicación del índice por una constante (b2*c2, b3*c2, b4*c2, b5*c2).
Cada vez que debamos modificar la lista de precios, sólo cambiamos el valor de la coordenada c1 para que SwiftCalc ponga en cada celda los nuevos valores.
Debido a esto también se efectúa una nueva suma, actualizándose el valor en b6.
En la figura 4 y 5 podemos ver la secuencia que representa lo recién explicado.
Estos ejemplos muestran la facilidad con que una hoja electrónica manipula los datos y ecuaciones numéricas.
Otra de las funciones que posee el SwiftCalc es la función MAX que determina el valor más grande que se encuentra en una columna o fila.
La figura 6 indica cada una de las funciones matemáticas del SwiftCalc.
Como si se tratase de un procesador de texto, esta planilla electrónica también posee búsqueda de datos o títulos, copiado y movimiento de celdas de un lugar hacia otro, salto a nueva celda y eliminación de celdas.
Por otro lado, SwiftCalc puede imprimir nuestra planilla sobre cualquier impresora Commodore.
Nosotros podemos indicarle, además, si se enviará ASCII normal o código Commodore, la dirección secundaria o SA y si requiere Linefeed después de cada impresión.
Los márgenes, tabulaciones y características de la impresión son también seleccionados desde esta opción.
En lo que respecta a la operación con disco, SwiftCalc puede almacenar y guardar nuestra planilla en disquete, ver el directorio o enviar comandos especiales, como el validate, new,etcétera.

TRUCOS

MUSICA PARA LA 64

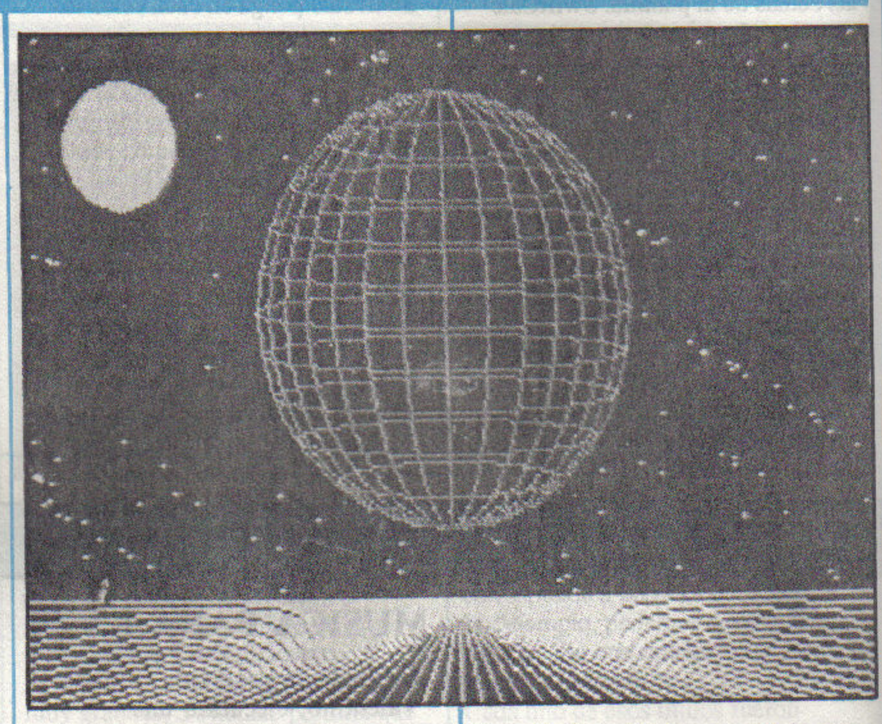
El siguiente programa, titulado Valentino, ejecutará una agradable melodía. Podrán utilizarlo como introducción a sus juegos o programas de aplicación.

```
10 PRINT"  VALENTINO"
20 FORJ=1TO154:READK:C=C+K:NEXT:IFC
   <>6245THENPRINT"ERROR EN
DATAS.":STOP
30 RESTORE:S=54272:V=S+24:FORJ=STOV
   1:POKEJ,0:NEXT:POKEV,15
40 READN,D:IFD=255THENPRINT"ACTIVADO
   !!!":END
50 POKES+1,N:POKES+5,3:POKES+15,3
60 POKES+4,17:FORDL=1TOD*250:NEXT
70 POKES+4,0:FORDL=1TO45:NEXT:GOTO40
71 DATA59,3,67,1,71,1,67,1,71,1,67,5
72 DATA59,2,67,1,71,1,67,1,71,1,67,6
73 DATA59,2,67,1,71,1,106,3,94,1,89,1
74 DATA79,7,71,1,67,1,106,3,71,1,71,2
75 DATA67,1,71,5,71,1,67,1,119,3,71,1
76 DATA71,2,67,1,71,5,71,1,67,1,134,3
77 DATA71,1,71,2,67,1,71,3,79,1,89,1
78 DATA119,5,71,2,79,1,89,1,79,1,89,1
79 DATA79,3,71,5,59,2,67,1,71,1,106,3
80 DATA94,1,89,1,79,7,59,2,67,1,71,1
81 DATA67,1,71,1,67,3,71,2,79,1,89,1
82 DATA79,1,89,1,79,3,71,2,79,1,89,1
83 DATA79,1,89,1,79,3,71,10,0,255
```

DIMENSION PASCAL

Con esta entrega culminamos la explicación de recursividad comenzada en el número anterior describiendo, además, los diferentes lenguajes Pascal que podemos encontrar para la Drean Commodore 64/C.

Como recordarán, en el número 16 comenzamos a explicar una de las características más "fuertes" del Pascal : la recursión.
A través de la función matemática factorial, ideal para explicar cómo opera la recursividad, describimos qué realiza internamente el lenguaje para concretar esta operación.
Ahora veamos la aplicación concreta. Cuando invocamos factorial (3), la función tiene que quedar pendiente para el valor 3, porque necesita saber cuánto es 2! (3!=3*2!) . En ese momento guarda el "estado" (variables, punto de interrupción de la función, etcétera) en el que se encontraba, para realizar la tarea que necesita y, una vez terminada, recuperar el "estado" y seguir adelante. Entonces, congela las cosas a nivel factorial (3) e invoca factorial (2). Nuevamente la función tiene que quedar pendiente porque hace falta saber cuánto es 1!. De manera similar se invoca factorial (1), llegando al final de la cadena, porque la función ahora es explícita: 1!=1. Cuando obtenemos el valor de 1! , corremos a dárselo a factorial (2), que estaba esperando ese valor. Cuando lo recibe, recupera el "estado" y calcula 2!.
Nuevamente, cuando obtenemos el valor de 2!, corremos para darle la noticia a factorial (3), que requería ese valor para su trabajo. Y finalmente factorial (3) devuelve su tarea hecha. Semejante desarrollo rara vez es posible desde BASIC y nunca tiene tanta elegancia.
La definición de procedimientos es similar a la de funciones, con la diferencia de que no nos devuelve valores (a menos que lo especifiquemos). Un procedimiento



puede ser, por ejemplo, la presenteción de un juego, imprimir algo de manera especial, etcétera..En los procedimientos también se pueden pasar parámetros.
Lo siguiente es un listado de código fuente correspondiente a un programa que calcula promedios, la parte entera del promedio y el factorial de la misma: (listado 1).
Read (variable) es la versión en PASCAL de Input. Write () es el análogo de Print. El agregado "ln" después de read y write hace que una vez realizada la operación se salte a la fila siguiente. El programa comienza a correrse a partir de la línea 1150 (mi Pascal tiene un editor no del todo ortodoxo), ya que lo que está antes entra en juego sólo cuando se lo invoca. El orden de un programa en Pascal es el siguiente: título, declaración de variables y sus tipos, declaración de procedimientos y funciones y, finalmente el programa principal. La versión de este programa en BASIC es la siguiente : (listado 2).
Como ven, un listado en PASCAL se lee mientras que uno en BASIC se intenta correr en la mente para ver qué hace. La ventaja de poder definir procedimientos y funciones que se agregan a las existentes es que facilita la comprensión del programa. Las instrucciones estructuradas aumentan el orden. La posibilidad de disponer las sentencias en la pantalla de la manera más caprichosa aumenta aún más ese orden. El programa deja de ser una tira de chorizos iguales; toma forma orgánica, sus partes se distinguen.Un algoritmo traducido de la mejor manera al BASIC sigue siendo incomprensible. Finalmente,

```
0000-1000 nombre del pograma
1010-1030 declaracion de variables
     generales del programa
1040-1080 procedimiento presentacion
1090-1140 funcion factorial
1150-1310 cuerpo principal del programa
```

```
0000-1000 va a subrutina presentacion
1010-1150 cuerpo principal del programa
1160-1180 subrutina presentacion
1190-1240 subrutina factorial
```

Listado 1

```
1000 PROGRAM PROMEDIO;
1010 VAR
1020  ACUMULADOR,AUX1,PROMEDIO,FACT:REAL;
1030  AUX2,NUMEROS,ENTERO:INTEGER;
1040 PROCEDURE PRESENTACION;
1050 BEGIN
1060  WRITELN('CUANTOS NUMERO QUIERE
      PROMEDIAR');
1070  READLN(NUMEROS);
1080 END;
1090 FUNCTION FACTORIAL(N:INTEGER):REAL;
1100 BEGIN
1110  IF N=1
1120   THEN FACTORIAL:=1
1130   ELSE FACTORIAL:=N*FACTORIAL(N-1);
1140 END;
1150 BEGIN
1160  PRESENTACION;
1170  ACUMULADOR:=0;
1180  FOR AUX2:=1 TO NUMEROS DO
1190   BEGIN
1200    WRITE('ENTRE EL NUMERO',
     AUX2,'   ');
1210    READLN(AUX1);
1220    ACUMULADOR:=ACUMULADOR+AUX1;
1230   END;
1240  PROMEDIO:=ACUMULADOR/NUMEROS;
1250  ENTERO:=TRUNC(PROMEDIO);
1260  FACT:=FACTORIAL(ENTERO);
1270  WRITELN('EL PROMEDIO ES ',PROMEDIO);
1280  WRITELN('LA PARTE ENTERA ES ',
     ENTERO);
1290  WRITELN('Y EL FACTORIAL DE LA PARTE
     ENTERA ES     ');
1300  WRITELN(FACT);
1310 END.
```

la declaración de variables obliga a sentarse y pensar el programa antes de tipear, desterrando las pruebas y los intentos de correr el programa "para ver qué pasa".

Y ahora...los defectos. Molesta mucho compilar el programa para darle los ajustes finales. Por mejor programador que usted sea, siempre va a tener algún error perdido que le va a llevar tiempo arreglar. Más tiempo que en escribir todo el programa. En general, cualquier programa en una hora se escribe y en una semana se depura.Depurar en Pascal implica compilación tras compilación sin posibilidad de un "trace" ni nada parecido. Una vez que el programa deja de correr no es posible ver el contenido de una variable como en BASIC. Otro defecto no atribuible al lenguaje es la ausencia de editores (en las máquinas chicas) que permitan el libre juego de la imaginación del usuario para encontrar su orden en el listado.

Para Commodore hay 3 versiones (las más difundidas) de Pascal. **Oxford Pascal** es un compilador para aprender. Manual muy lindo, con muchos ejemplos, flojo en algunos temas (punteros) y muy flojos en otros (no hay número de página, sino número de sección). Las instrucciones gráficas no son muy buenas, pero permiten, al menos, hacer puntos y líneas. También puede trabajarse con pantalla partida (gráficos y texto) . Tarda bastante en cargar y compila un poco lentamente. La edición de textos se realiza a través de líneas (como en Basic), lo que muchas veces impide realizar un código fuente entendible; otra limitación en este sentido es el hecho de trabajar con 40 columnas solamente. Permite crear programas ejecutables en el disco para no tener que compilarlos cada vez que los querramos usar, y también se puede crear versiones de los programas que corren independientemente del compilador.

El **Super Pascal** (versión para 64) adolece de los mismos problemas de edición de su émulo Oxford. Pero esta vez aparece como un archivo más un código fuente que, compilado, da origen al editor. Con

un poquito de paciencia y ganas, podemos llegar a entender cómo funciona un editor de textos y, a partir de eso, agregar comandos e instrucciones para imitar la edición de las máquinas más grandes (todavía no he visto ninguna manera de hacer un scroll del listado para abajo y para arriba; sería una forma mucho más fácil de acceder a una línea de programa, que hacerlo a través de un list 567, por ejemplo). El compilador está, en general, mucho más organizado que el Oxford. A partir de un menú principal, vamos al editor, escribimos nuestro programa, salimos rapidamente de él al menú principal, y elegimos la opción de compilado. Se acompaña el programa con un paquete utilitario para manejo de disco, otro que contiene el código fuente de las rutinas gráficas, y un ensamblador para mezclar Pascal y código de máquina. Un defecto algo doloroso es el ruido que emite nuestra disquetera al compilar un programa en el disco. Hay una versión de Super Pascal para 128.

Finalmente para 128 en CP/M está el **Turbo Pascal** (versión CP/M 80). El mayor problema es la ausencia de instrucciones gráficas.Tanto para programación educativa como para la científica puede llegar a representar serios problemas. Si bien un programa serio no requiere muchos colores ni sprites con forma de duende, graficar algún tipo de información o realizar alguna tarea en la que los gráficos sean pocos pero fundamentales, se hace imposible. El manual es gordo pero no enseña; se dedica a explicar las instrucciones y las características para cada sistema (las otras configuraciones del compilador están pensadas para CP/M 80, CP/M 86 MS-DOS). Como se entenderá, dado que la mayoría de los programas se encuentra corriendo en las IBM-PC, las rutinas gráficas y de sonido se hallan sólo en las versiones para esa máquina; agregar rutinas gráficas para el CP/M 80 implicaría programar versiones de esas rutinas para todos los procesadores de video de todas las máquinas que usasen CP/M 80 dado que eso es imposible, nos quedamos sin rutinas gráficas. El resto de los problemas que plantea el uso de este compilador son atribuibles al CP/M de la 128. Recién en las últimas versiones del sistema se encuentra la repetición de tecla automática, no muy rápida, pero más conveniente que tener que recorre 30 caracteres a "dedo". Muchos de los que queremos programar en serio lloramos la carencia de un monitor "de 80". No sé por qué, y espero que alguien tenga la respuesta, en un monitor de 80 columnas (obviamente trabajando en ese modo) el manejo de pantalla del CP/M es más rápido. La conclusión general implica, en particular, que el manejo de pantallas que hace el Turbo es más rápido también, por lo que la edición se hace tan fácil que uno no sabe si está trabajando con una PC o con una home computer. Claro, volvemos a la tierra cuando queremos hacer un círculo o dibujar una raya : por el resto, es un Pascal en serio. No tiene número de línea en el listado, por lo que éste debe ser manejado como si fuese un libro (con páginas); la edición del programa tiene comandos muy poderosos (inclusive con la misma sentencia de teclas que en el WordStar); y la compilación es mucho más rápida que la de los engendros anteriores. Para terminar, un lenguaje es un conjunto de signos con reglas que sirve para comunicar cosas. En tanto mejor se adapte el lenguaje a los interlocutores, la comunicación es más fácil. Desgraciadamente para BASIC, los terrestres no entendemos bien cosas como gosub 1234 o barbaridades gastronómicas como fideos "goto", nombre popular con que se identifica a los programas spaguetti de los socios más activos del BASIC's CLUB.

ALEJANDRO PARISE

Listado 2

```
1000 GOSUB 1160:REM PRESENTACION
1010 AC=0
1020 FOR A1=1 TO NU
1030 PRINT "ENTRE EL NUMERO ";A1;
1040 INPUT A2
1050 AC=AC+A2
1060 NEXT
1070 PR=AC/NU
1080 EN=INT(PR)
1090 T=EN:GOSUB 1190:REM FACTORIAL-
     T ES LA VARIABLE DE ENTRADA
1100 FA=U:REM U ES LA VARIABLE DE SALIDA
1110 PRINT "EL PROMEDIO ES ";PR
1120 PRINT "LA PARTE ENTERA ES ";EN
1130 PRINT "Y EL FACTORIAL DE LA PARTE
     ENTERA ES  "
1140 PRINT FA
1150 END
1160 PRINT "CUANTOS NUMEROS QUIERE
     PROMEDIAR ";
1170 INPUT NU
1180 RETURN
1190 I=1
1200 FOR G=1 TO T
1210 I=I*G
1220 NEXT
1230 U=I
1240 RETURN
```

CLUB DE USUARIOS

Nuestro nuevo boletín

Finalmente ya tenemos listo el boletín del Club de Usuarios Drean Commodore. En él encontrás respuestas a todas tus dudas.
A través de las inquietudes que durante este tiempo nos han acercado nuestros socios, hemos elaborado una veintena de preguntas junto a sus respuestas que reflejan los interrogantes de muchos de los usuarios de Drean Commodore.
Este nuevo boletín podrás usarlo como un pequeño manual de consulta, que te ayudará a evacuar cualquier duda.

La mejor interfase para el mejor club

Esta es una de nuestras novedades. Somos distribuidores exclusivos de la nueva interfase paralelo Solter.
Sin lugar a dudas, es una de las mejores interfases centronics del mercado actual.
Gracias a su revolucionaria tecnología, este nuevo dispositivo nos permite conectar nuestra Drean Commodore 64/C o Commodore 128 a cualquier impresora con entrada paralelo, incluyendo la MPS 1000 y la nueva MPS 1200.
A diferencia de otras interfases de su tipo, Soltec se alimenta directamente desde la línea de 220 Volts.
A través de un circuito de protección, se aisla totalmente la computadora de la alimentación, impidiendo sobrecargas a la parte lógica y de control.
Internamente dispone de un amplio buffer que se alimenta a velocidades altísimas, permitiendo que la impresora desarrolle su máxima velocidad de impresión.

Cursos vía Delphi

Aquellos socios que estén suscriptos a Delphi y quieran conocer más sobre nuestros cursos, encontrarán en nuestra sección (grupos y clubes) toda la información detallada de los cursos que el Club está dictando.
Por otro lado, continuamos dictando los cursos de aplicaciones específicas para profesionales.
No hemos descuidado nuestros cursos tradicionales que venimos dictando en forma periódica.
Así, mes a mes, cientos de usuarios concurren al Club para asistir a los cursos de caracter general (Introducción a la informática, Lenguajes, etcétera) como así también los de índole específica.

CP/M al por mayor

Seguimos recolectando programas, lenguajes y utilitarios para la Commodore 128, que trabajan bajo el popular sistema operativo CP/M.
Cada programa se entrega con su correspondiente manual.
Entre algunos de ellos podemos citar el C-COMPILER, COBOL, D-BASE II COMPILER, DBASE II, DUTIL, FORTRAN 80, LISP, M-BASIC COMPILER, MEX.COM TERMINAL (software de comunicaciones), MS-BASIC, MULTIPLAN, PERFECT WRITER, PROLOG, QUICKCODE, TURBO PASCAL y WORDSTAR (adaptado para la MPS 1000).
Los interesados en adquirir algunos de estos programas deberán dirigirse a nuestra sede, sita en Pueyrredón 860, 9º piso, Capital Federal.

Manuales

El Club de Usuarios Drean-Commodore dispone en la actualidad de una cantidad considerable de manuales a disposición del público en general. En esta edición comenzamos a publicar la primera parte del listado de dicho material.

Num.	Título	Tipo	Equipo
31	Basic Microsoft	Idioma	C-128
60	Bgraph 64	Utilitario	C-64
26	Cad de Abacus	Utilitario	C-128
48	Cad Pack 128	Utilitario	C-128
44	Calk Kit	Utilitario	C-64
55	Cobol 128	Idioma	C-128
19	Cobol de Microsoft	Idioma	C-128
17	Compilador C-128 Abacus	Utilitario	C-128
21	Curso de Basic (I)	Educativo	C-64
21	Curso de Basic (II)	Educativo	C-64
45	D-File	Utilitario	C-128
34	Data Manager	Utilitario	C-128
49	Data Manager	Utilitario	C-64
25	DBASE II (aplique)	Utilitario	C-128
50	Desk Manager	Utilitario	C-64
36	Di-Sector	Utiltario	C-64
12	Easy Mail	Utilitario	C-64
43	Easy Script	Utilitario	C-64
29	Font GEOS	Utilitario	C-64
67	Fortran 80	Idioma	C-128
32	GEOS	Utilitario	C-64
53	GEOS	Utilitario	C-64
7	Guía de referencia de C64	Utilitario	C-64
4	Guía del datasete 1530	Utilitario	1530
5	Guía del program. 128 (I)	Utilitario	C-128
5	Guía del program. 128 (II)	Utilitario	C-128
5	Guía del program. 128 (III)	Utilitario	C-128
23	Guía del usuariuo de C-64	Utilitario	C-64
2	Guía usuario CP/M C-128	Utilitario	C-128
3	Guía usuario de 1541	Utilitario	1541
51	Jane	Utilitario	C-128
41	Kawasaki sinthesizer	Música	C-64
9	LOGO Drean	Idioma	C-64
37	Magic Desk	Utilitario	C-64
58	Manual de 1541	Utilitario	1541
57	Manual de MPS-803	Utilitario	MPS-803
54	Manual de MPS-1000	Utilitario	MPS-1000
6	Manual del usuario1571	Utilitario	1571
1	Manual del usuario de 128	Utilitario	C-128

LAS COMPUTADORAS QUE PIENSAN

Un programa desarrollado por el Club de Usuarios les permite a las computadoras aprender, descubrir errores y mantener una lúcida conversación con los seres humanos.



El desarrollo de la "inteligencia artificial", es decir un computador que emule los procesos del pensamiento humano, es uno de los mayores desafíos que enfrenta la informática, ambición que, de la mano de los llamados "programas heurísticos" parece en camino de concretarse parcialmente.

Estos programas, caracterizados por incorporar a su "banco" cada nuevo dato que se les suministre, simulan con bastante exactitud los mecanismos mentales del hombre y, quizás, la única y tenue —pero primordial— demarcación que los separa aún de la verdadera inteligencia, es su falta de "conciencia", aunque la real definición de este concepto ya ingresa en los campos de la filosofía, la ética y la moral.

El término de programas "Heurísticos" se difundió mundialmente de la mano de la novela de ciencia ficción escrita por Arthur Clark "2001, Odisea del Espacio", llevada a la pantalla por el director Stanley Kubrik.

El "cerebro electrónico" de la nave "Descubrimiento", y que finaliza en un estado de psicosis, tenía como bautismo "HAL 9.000" sigla que significaba —según explica la novela de Clark— computador programado heurísticamente.

GRADOS DE "INTELIGENCIA"

Al igual que en los seres humanos, el grado de "inteligencia" de un computador está directamente vinculado con su capacidad "cerebral" o, en este caso concreto, su "memoria", interna o externa.

No obstante los computadores hogareños Drean-Commodore están en condiciones de trabajar con programas heurísticos, en lenguaje Prolog o Lisp, que simulan con un grado asombroso de realismo (para los legos), una conversación inteligente y que les permite en cada diálogo "aprender", es decir incorporar a su "archivo" nuevos conceptos.

Este es el caso de "Aprende", soft desarrollado por el Club de Usuarios Drean-Commodore en base a distintos programas procedentes del extranjero.

Su "método de trabajo" es muy simple: la Drean-Commodore le pide a su interlocutor que piense en un animal y luego comienza con una serie de preguntas destinadas a definirlo.

Así, paso a paso lo interrogará acerca de si vive en el agua o en tierra, si es mamífero u ovíparo, carnívoro o vegetariano y otros datos que le permitan llegar a una conclusión acerca de cuál es el animal seleccionado.

Hasta aquí el proceso no resulta en exceso sorprendente, pero puede ocurrir que más de un animal reúna las características solicitadas por la computadora y que ésta se equivoque.

Ante el error, el programa solicita a su interlocutor que le mencione las características que diferencian al animal elegido del citado, y automáticamente lo incorpora a su programa, es

decir que la máquina "aprendió", en un proceso de selección similar al de un chico en la escuela.
En este caso, obviamente, el programa carece de conciencia, y "camina" a ciegas por el árbol de conocimientos previamente suministrado hasta llegar a un resultado "inevitable" por el proceso de desechar otras opciones.
El efecto de simulación de inteligencia ocurre en el momento en que la máquina "descubre" su error y solicita los datos que le posibilitan evitar su reiteración, los que se incorporan a su software básico.
Los programas heurísticos permiten asimismo, por ejemplo, descubrir errores de programación y aceptarlos, como el caso en que, al suministrar instrucciones en Basic, se ordena a la máquina "Pront" en lugar de "Print".

"SISTEMAS EXPERTOS"

Estos programas heurísticos que simulan inteligencia son conocidos bajo la denominación de "sistemas expertos" y, aplicados a ramas tan diferentes como la explotación petrolera, meteorológica o la medicina, brindan grandes ayudas y hasta posibilitan distintos descubrimientos mediante un proceso de aprendizaje y deducción.
En Estados Unidos es común la existencia de grandes computadores con una base de datos que interroga a cada médico acerca del estado de un determinado paciente y sus síntomas, para ayudarlo en el diagnóstico de cada enfemedad.
Sin embargo, hubo casos en que la totalidad de los datos aportados no "encajaba" en los conocimientos previos del computador ni en los elementos adicionales que pudo brindarle un experto humano u otro sistema informático experto al que se consultó, lo que llevó al sistema a la conclusión —corroborada luego por un científico— de la existencia de algún nuevo tipo de virus o enfermedad.
En la actualidad en Japón se diseñan sistemas expertos de gran rendimiento para Personal-Computers (PC) e incluso ya los computadores hogareños Drean-Commodore están en condiciones de trabajar con estos programas heurísticos, tal como lo demuestra "Aprende".
Su capacidad de trabajo es más limitada, en el caso de "Aprende" es capaz de almacenar los datos para seleccionar por ejemplo un millar de animales diferentes. Pero son capaces de brindar aplicaciones prácticas en tanto y en cuanto se limite su campo de acción para no "desbordar" la memoria.

TRUCOS

KEYPAD

Los usuarios de la Commodore 128 sabrán que cuando pasan a modo 64, el teclado numérico (Keypad) no está conectado en dicho modo.
Seguramente coincidirán en la practicidad del Keypad. Para que puedan seguir utilizándolo en modo 64, les presentamos el siguiente programa.
Este corre en modo 64 y permite detectar cuando se oprime algunas de las teclas del Keypad.
Carguen y ejecuten el programa. En caso de cometer algún error en las sentencias DATAs, el programa lo informará.
Tengan cuidado que se autoborra. Por ello asegúrense de grabarlo en disco o en cassette.

```
10 REM KEYPAD EN MODO 64
20 REM CUIDADO QUE SE AUTOBORRA
100 FORA=830TO949:READ B:C=C+B
110 POKEA,B:NEXT
120 IFC<>13039THENPRINT"ERROR EN DATAS.
   VERIFICA LOS
VALORES":STOP
130 SYS 830:PRINT"KEYPAD ACTIVADO":STOP
140 NEW
150 REM VALORES EN DECIMAL
160 REM DEL PROGRAMA EN LM
170 REM ************************************
830 DATA 120,169,75,141,20,3,169,3,141,21
840 DATA3,88,96,169,248,141,47,208,169,255
850 DATA141,0,220,205,1,220,208,10,141,47
860 DATA208,74,141,0,220,76,49,234,160,0
870 DATA140,141,2,169,251,141,47,208,162,0
880 DATA173,1,220,205,1,220,208,248,74,144
890 DATA9,200,202,208,249,110,47,208,176,234
900 DATA185,157,3,16,7,162,1,142,141,2
910 DATA41,127,133,203,169,255,141,47,208,32
920 DATA72,235,76,126,234,64,35,44,135,7
930 DATA130,2,64,64,40,43,64,1,19,32
940 DATA8,64,27,16,64,59,11,24,56,64
950 REM ************************************
```

COMO TRABAJA EL INTERPRETE BASIC

Las posibilidades de la 64-C parecen infinitas. Para demostrarlo les explicaremos cómo es posible agregar más comandos y sentencias a la computadora, provocar una interupción y traducir los errores y sentencias al castellano.



Hoy en día es muy sencillo adquirir una computadora que tenga comandos e instrucciones que permitan hacer lo imposible.
De acuerdo a comerciales publicitarios, muchas de esas computadoras nos permiten trabajar, entre otras cosas, "con 300.000 colores, 1000 tipos de letras distintas y generar voz marciana".
Pero no todas permiten que su Basic residente sea ampliado por el usuario, es decir que nosotros podamos diseñar e implementar nuevas instrucciones, como es el caso de la Drean Commodore 64/C.
Gracias al diseño del sistema operativo de esta computadora, no sólo podemos agregar más comandos y sentencias sino que, además, podemos ejecutar procesos en intervalos de tiempos periódicos, provocar una interrupción al presionar una tecla y hasta traducir los errores y sentencias al castellano.
Por supuesto que para realizar todas estas modificaciones es necesario disponer de una documentación precisa, lo cual no representa ningún problema para la Drean Commodore 64/C ya que podemos encontrar en el mercado bibliografía que nos explica desde cómo está formado el teclado de la Drean Commodore 64/C hasta cómo y dónde están las rutinas que utiliza el intérprete Basic.
Este último tema será el que trataremos, explicando cómo el programador puede acceder a las rutinas del sistema operativo que efectúan distintas tareas.
Veamos por ejemplo la rutina que busca dentro de un programa Basic una determinada línea.
Esta rutina, que se denomina FNDLIN (del inglés FIND LINea-buscar línea), se usa para transferir el control del programa a través del GOTO o GOSUB e imprimir la línea solicitada por el LIST.
Pero antes de explicarla veamos cómo se almacena en la memoria el programa que nosotros hacemos.
Cada sentencia (PRINT, AND, WAIT, etcetera) se codifica a un número de un byte denominado "token". Este es único, de manera que no hay un mismo token que corresponda a dos comandos distintos.
Esta codificación se lleva a cabo por dos motivos :
1) Ahorro de memoria : En vez de utilizar 5 direcciones de memoria para almacenar los caracteres de, por ejemplo, PRINT, se utiliza un solo

Figura 1

MEMORIA RAM

Dirección	Contenido	
$0801	0C	Puntero a dirección próxima sentencia ($080C)
$0802	08	
$0803	0A	Número de línea
$0804	00	
$0805	99	Token
$0806	.	Texto vario
$0807	.	
$0808	.	
$0809	.	
$080A	.	
$080B	00	Fin de línea
$080C	.	

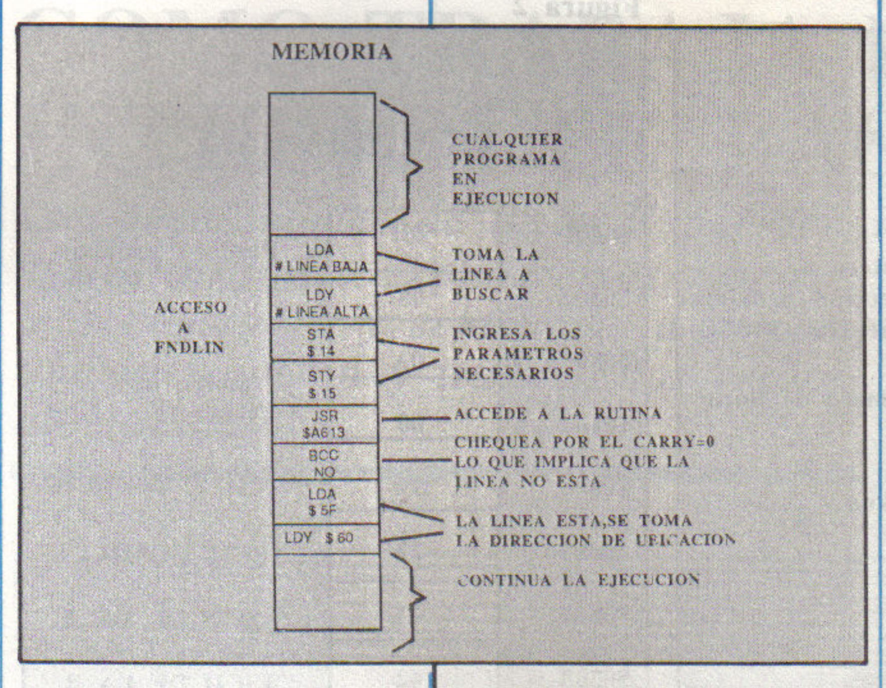
Figura 2

MEMORIA RAM

Dirección	Contenido	
$0801	0D	Puntero próxima sentencia ($080D)
$0802	08	
$0803	0A	Número de línea 10
$0804	00	
$0805	99	Token de PRINT
$0806	22	ASCII COMILLAS
$0807	41	ASCII DE LA A
$0808	42	ASCII DE LA B
$0809	43	ASCII DE LA C
$080A	22	ASCII DE LAS COMILLAS
$080B	00	
$080C	.	

byte (su token).
2) Velocidad de traducción:
Siguiendo en el caso de PRINT, el intérprete sabe que se trata de esta sentencia con sólo ver el token.

Si se hubiese optado por almacenar los cinco caracteres, el intérprete sabe que se trata de PRINT luego de comparar el segundo caracter.
Es decir que cada línea Basic que constituye nuestro programa se almacena como se indica en la figura 1.
Los primeros dos bytes indican la dirección en donde se encuentra la




```
$A613 LDA $2B        $A621 INY            $A62C BNE $A637      $A638 LDA ($5F),Y
$A615 LDX $2C        $A622 INY            $A62E LDA $14        $A63A TAX
$A617 LDY #$01       $A623 LDA $15        $A630 DEY            $A63B DEY
$A619 STA $5F        $A625 CMP ($5F),Y    $A631 CMP ($5F),Y    $A63C LDA ($5F),Y
$A61B STX $60        $A627 BCC $A641      $A633 BCC $A641      $A63E BCS $A617
$A61D LDA ($5F),Y    $A629 BEQ $A62E      $A635 BEQ $A641      $A640 CLC
$A61F BEQ $A640      $A62B DEY            $A637 DEY            $A641 RTS
```

próxima línea Basic.
Los siguientes dos bytes representan el número de línea.
Finalmente el quinto byte indica el token, es decir de qué sentencia se trata.
El "0" que se ve al final sirve para indicarle al intérprete que allí termina esa línea.
Como ven, el número de línea se representa a través de dos bytes, utilizando el formato bajo-alto.
A través de él se puede representar cualquier número entero comprendido entre 0 y 65535. Es por eso que el máximo número de líneas que nuestro programa puede tener es, justamente, 65535.
Cuando nosotros ingresamos una línea Basic como :
10 PRINT "ABC"
el intérprete la almacena en memoria con el formato representado en la figura 1.
Para convertir el número de línea al formato antes citado, el intérprete realiza la siguiente operación. La parte alta se calcula como :
PA=INT(NUMERO/256)
mientras la parte baja se determina a través de :
PB=NUMERO-256*PA
La variable NUMERO representa al número de línea ingresado.
Finalmente convierte PA y PB en números hexadecimales y los almacena.
Luego toma la sentencia (en nuestro caso PRINT) y se fija qué token le corresponde, para luego ponerlo en la memoria.
Finalmente toma lo que está entre comillas almacenando cada uno de los caracteres en su correspondiente código ASCII.
La figura 2 indica cómo queda en memoria la línea 10.
La rutina que busca en el programa Basic el número de línea indicado va comparando los bytes que indican el número de línea actual con el cue se debe buscar.
Si lo encuentra pone la dirección de comienzo, es decir la ubicación en la memoria en donde está almacenada, en dos direcciones consecutivas.
Como cada dirección está formada por 16 bits, es decir 2 bytes, es necesario utilizar dos direcciones de memoria para poder almacenar semejante valor.
No olviden que en una dirección podemos poner, a lo sumo, valores comprendidos entre 0 y 255, es decir un byte por dirección.
Un pequeño resumen de la operación de esta rutina sería :
1) Leer la dirección a buscar desde las direcciones hexadecimales 14 y 15 (normalmente estas direcciones sirven para múltiples propósitos) .
2) Iniciar la busqueda .
3) Si la línea no está en el programa, pone el flag de carry a "0" y finalizar la búsqueda.
4) Si la línea está, pone el carry a "1" y la dirección de comienzo de almacenamiento en las direcciones $5F y $60 .
Para los curiosos les dejamos el listado 1 que corresponde a la rutina descripta .
Recuerden que al finalizar la ejecución de la misma, y si la línea está en el programa Basic, podemos ir a buscar la dirección en donde esa línea está en $5F y $60 (parte baja y alta respectivamente) .
Para utilizar esta rutina debemos hacer :
1) Poner en las direcciones $14 y $15 el número de línea a buscar respetando el formato alto-bajo .
2) Acceder a la rutina FNDLIN a través de JSR $A613 (en esta dirección se encuentra almacenada) .
3) Ver cómo está el flag de carry teniendo en cuenta que :
3.1) Si está a "0" , la línea no se encuentra en la memoria.
3.2) Si está en "1" , la línea está y su dirección de almacenamiento en la RAM se toma desde las direcciones $5F (parte baja) y $60 (parte alta) .
La figura 3 describe todos los pasos que se realizan durante la ejecución de esta rutina .
En el próximo número continuaremos describiendo otras rutinas que se emplean en la ejecución de nuestros programas Basic .

DEFINIDOR DE TECLAS DE FUNCION



Comp.:* *Drean Commodore 64/C
Conf.:* *Básica
Tipo:* *Utilitario

Este programa, elaborado por el Club de Usuarios Drean Commodore, carga una rutina en lenguaje de máquina que nos permite definir fácilmente las teclas de función.

Luego de tipear y ejecutar el programa, éste nos dirá qué debemos hacer para reactivarlo.

Si las DATAs fueron correctamente cargadas, el programa estará listo para definir las teclas de función.

A partir de aquí podemos comenzar a definir cada una de las ocho teclas de función.

Sólo basta indicar la tecla de función y el comando que se le asignará.

Así, cada vez que presionemos esa tecla, se ejecutará el comando estipulado.

Por ejemplo al tipear F1=LIST y luego RETURN, otorgaremos a la tecla de función 1 el comando LIST. Cada vez que la presionemos, ejecutaremos este comado.

Variables utilizadas

Nombre	Descripción
I	Indica dirección de carga
CK	Chek sum de los valores cargados
A	Representa las instrucciones que se POKEarán en la memoria
SH,SL	Intervienen en el cálculo do la dirección de reactivación del programa.

Descripción del programa

Línea	Descripción
100	Determina la dirección inicial de carga
110	Indica inicio de carga de la rutina en memoria
120	POKEa en memoria las instrucciones
170	Controla si los valores cargados son
180	Aciva la rutina
230-820	Instrucciones de la rutina codifica en decimal

```
10 REM**********************************
20 REM*                                *
30 REM*        CLUB DE USUARIOS        *
40 REM*        DREAN  COMMODORE        *
50 REM*                                *
60 REM* DEFINIDOR DE TECLAS DE FUNCION *
70 REM*                                *
80 REM**********************************
90 REM  F1=LIST <RETURN>. AL PRESIONAR
   F1, APARECE LIST EN LA PANTALLA
100 I=49152
110 PRINT:PRINT:PRINTTAB(10)"UN MOMENTO
   POR FAVOR"
120 FORX=1TO129:READA:I=I+1:POKEI,A:NEXT
130 SH=INT(I/256):SL=I-SH*256
140 READA:CK=CK+A:IFA=256THEN170
150 IFA<0THEN200
160 POKEI,A:I=I+1:GOTO140
170 IFCK<>28195THENPRINT"ERROR EN LOS
   DATA":STOP
180 SYS(SH*256+SL+403)
190 PRINT:PRINT:PRINTTAB(8)"USE SYS"SH*
   256+SL+403"PARA REACTIVAR":NEW
200 IFA<-255THENA=ABS(A+256)+SH:GOTO160
210 A=ABS(A+1)+SL:GOTO160
220 IFA<0THEN200
230 DATA173,-1,-256
240 DATA240,30,166,198,224,11
250 DATA176,24,168,185,-1,-256
260 DATA240,15,157,119,2,230
270 DATA198,238,-1,-256,173,-1
280 DATA-256,41,15,208,3,141
290 DATA-1,-256,108,-146,-257,32
300 DATA72,235,173,-1,-256,208
310 DATA34,165,157,240,30,165
320 DATA212,208,26,166,198,202
330 DATA189,119,2,201,133,144
340 DATA16,201,141,176,12,56
350 DATA233,133,10,10,10,10
360 DATA9,1,141,-1,-256,96
370 DATA166,122,189,0,2,201
380 DATA70,208,68,232,189,0
390 DATA2,201,49,144,60,201
400 DATA57,176,59,41,15,168
410 DATA135,-135,-257,168,232,189
420 DATA0,2,201,61,240,5
430 DATA162,11,108,0,3,232
440 DATA189,0,2,201,13,240
450 DATA15,201,95,208,2,169
460 DATA13,153,-1,-256,200,152
470 DATA41,15,208,233,169,0
480 DATA153,-1,-256,160,107,32
490 DATA47,241,108,2,3,108
500 DATA-144,-257,208,251,232,189
510 DATA0,2,201,61,208,202
520 DATA232,138,72,160,0,189
530 DATA0,2,201,44,240,8
540 DATA232,200,192,15,208,243
550 DATA240,182,192,0,208,4
560 DATA162,8,208,176,232,189
570 DATA0,2,201,49,240,8
580 DATA201,56,240,4,162,9
599 DATA208,160,41,15,170,152
600 DATA72,160,0,138,32,186
610 DATA255,104,168,104,170,152
620 DATA160,2,32,189,255,162
630 DATA-1,134,251,169,-256,133
640 DATA252,160,-257,169,251,162
650 DATA-227,32,216,255,169,13
660 DATA32,210,255,76,116,164
670 DATA0,1,65,17,81,33
680 DATA97,49,113,124,165,49
690 DATA234,162,-256,173,5,3
700 DATA201,-256,240,17,141,-145
710 DATA-257,173,4,3,141,-144
720 DATA-257,169,-211,141,4,3
730 DATA142,5,3,173,21,3
740 DATA201,-256,240,19,141,-147
750 DATA-257,173,20,3,141,-146
760 DATA-257,169,-130,120,141,20
770 DATA3,142,21,3,88,173
780 DATA144,2,201,-256,240,19
790 DATA141,-170,-256,173,143,2
800 DATA141,-169,-256,169,-168,120
810 DATA141,143,2,142,144,2
820 DATA88,96,256
```

HARRIER



Rating Total: B
Creatividad: B
Profundidad del juego: B
Valor en relación al precio: Se justifica
Computadora: Drean Commodore 64/C
Editor: Mirrorsoft

En este juego debemos comandar un caza bombardero Harrier equipado con las más alta tecnología aeroespacial.
De apenas unos 200 bloques, en este nuevo juego tenemos que destruir al enemigo usando al avión antes mencionado.
La nave dispone de despegue vertical y del tipo catapulta. Igualmente podemos despegar en la forma tradicional, es decir efectuando un carreteo de despegue común.
La primera pantalla en aparecer corresponde a la presentación del juego, lograda a través de un excelente gráfico del Harrier.
En la siguiente pantalla, el programa nos pide que seleccionemos el tipo de juego a desarrollar.
Este puede ser Practice, Combat, Demo y Combat Practice. Para cada una de éstas podemos seleccionar el nivel de juego que puede ser Pilot (fácil), Commander (mediano) y Ace (difícil).
A través del joystick conectado en la puerta 2 seleccionamos la dificultad de nuestra misión.
A partir de aquí la cosa se pone difícil. Dentro de la cabina y con luz verde para despegar, tendremos que manejar hábilmente los controles de la nave para no estrellarnos sin haber iniciado el combate.
Las teclas "P" y "O" controlan la aceleración de la nave. La primera de ellas la aumenta mientras que la segunda realiza lo contrario.
Podemos decir que son las teclas que más se usan durante el desarrollo del juego ya que HARRIER, y a diferencia de otros, simula muy bien ciertas condiciones de vuelo como descenso en picada (el juego nos advierte que la estructura del avión no soportará la aceleración) o ascenso con poca propulsión.
El comando del avión se logra a través del joystick y, por supuesto, con el botón se lanzan las bombas o se acciona la ametralladora.
El despegue podemos efecuarlo de tres formas distintas: vertical, horizontal o modo catapulta.
Con las teclas "3", "4" y "5" seleccionamos el modo de despegue. Para los novatos como nosotros, les aconsejamos que despeguen horizontalmente; es más fácil.
El radar de lucha nos indica nuestra posición y la del enemigo. Podemos optar por destruir bases terrestres o subir a 10.000 pies para iniciar una "lucha de perros" con los jets enemigos.
En ambos casos, HARRIER dipone de una mira automática que marca al objetivo a destruir siguiéndolo por todos lados. De este modo, el misil que disparemos seguirá y destruirá al enemigo. No hay manera alguna de esquivar la mira automática.
Si por el contrario somos objetivos de los misiles enemigos, la computadora de abordo nos lo avisará. Disponemos de unos pocos segundos para lanzar los "flares", quienes se encargarán de despistar al sistema de guía del misil.
Otros de los comandos del HARRIER son la tecla "U" que levanta o baja el tren de aterrizaje, la letra "A" que indica la cantidad de blancos acertados, "S" que deja o quita el sonido de las turbinas, "F" que habilita los flaps y "B" que se encarga de activar y desactivar los frenos aerodinámicos.
Todas estas teclas disponen de su correspondiente indicador ubicado sobre el tablero de la nave.
Estas, junto al estabilizador, control de combustible/aceleración, radar de combate y mapa de ubicación, contribuyen a que la simulación de la lucha en HARRIER sea de lo mejor.

FIGHTING WARRIOR



Rating Total: B
Creatividad del juego: B
Profundidad: B
Valor en relación al precio: Se justifica
Computadora: Drean Commodore 16
Editor: Melbourne House
Distriubuidor oficial: Graphic Games

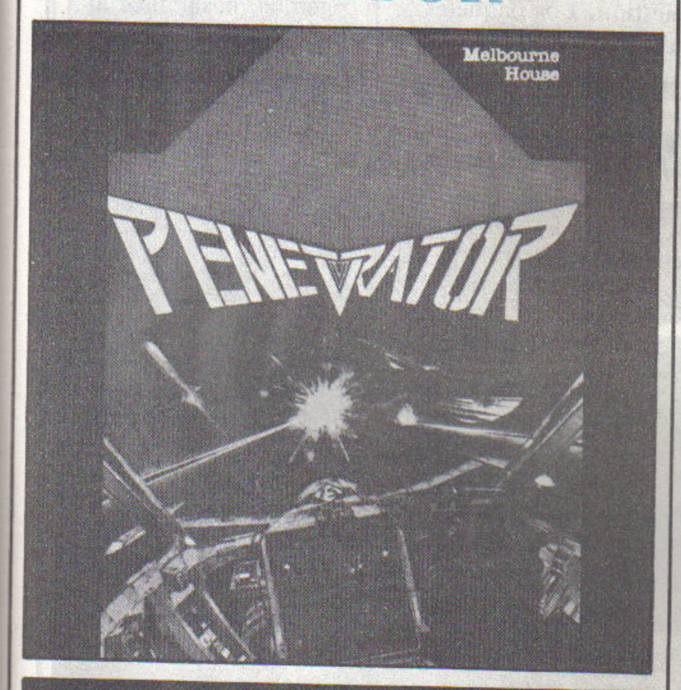
REVISIÓN DE SOFTWARE

Nuestra misión en FIGHTING WARRIOR consiste en rescatar a la princesa Susy, quien está bajo el poder del temible Kuky, rey de los truhanes.

El acceso a la princesa Susy no es fácil. Los secuaces de Kuky tratarán de matarnos antes de que el juego comience.

A lo largo del juego veremos que nuestros movimientos quedarán bloqueados por una señal. Tendremos que aniquilarla, a través de un certero golpe. Luego podremos seguir la búsqueda de Susy.
Podemos utilizar el teclado o el joystick conectado en el port 1.
Si usamos el teclado como control del juego, deberemos usar las teclas Q, Z, I, P.
Cada una de ellas se encarga de movernos hacia arriba, abajo, izquierda, derecha.
Con la barra espaciadora dispararemos.

PENETRATOR



Rating Total: B
Creatividad del juego: B
Profundidad: B
Valor en relación al precio: Se justifica
Computadora: Drean Commodore 16
Editor: Melbourne House

La acción transcurre en el espacio, combatiendo contra las fuerzas malignas del coronel Guey.
A la nave la podemos controlar fácilmente con las teclas Q (arriba) y Z (abajo). La barra espaciadora es quien se encarga de soltar los láser.
Nuestra misión consiste en recoger a un piloto aliado en un mundo desconocido.
Las fuerzas del coronel Guey nos estarán esperando para matarnos. Sólo podremos aterrizar cuando acabemos con todos ellos.
El secreto de la misión reside en ser lo más rápido posible en acabar a los soldados de Guey para así poder rescatar al piloto amigo.

DECEPTOR



Rating Total: B
Creatividad: B
Profundidad del juego: B
Valor en relación al juego: Se justifica
Computadora: Drean Commodore 64/C
Editor: Avantage

Casi todos conocemos los nuevos "dibujitos" animados que pasan por la tele llamados Transformers".
Para los que desconocen esta serie les contamos que se trata de un grupo de robots "buenos" que se pueden transformar en vehículos, aviones, etcétera para combatir al robot "malo".
DECEPTOR es un juego en el que debemos manejar un robot el cual se puede convertir en auto, jet o robot lanza laser.
La presentación es realmente muy buena, lograda con unos gráficos bien hechos y, como siempre, hechos gracias a la capacidad gráfica de la 64/C.
La música del juego también está bien lograda.
El juego comienza con su correspondiente presentación (un brujo con el puño cerrado diciéndonos barbaridades irrepetibles), dándonos los tipos de acción que podemos desarrollar.
En esta parte se nos permite practicar antes de comenzar a jugar. Esto nos permite controlar y ver cómo se hacen las transformaciones del robot XTU - + 657 que, como dijimos, puede ser un auto, avión o robot lanza laser.
A través de joystick debemos guiarlo por el interior de la base del Dragón Rojo, formada por rampas, campos eléctricos, víboras y gotas de lluvias mortales.
El Dragón tratará a toda costa de aniquilarnos utilizando a sus secuases.
Si creemos que ya estamos listos para jugar por puntos, volvemos al menú principal para seleccionar la segunda opción.
Aquí la cosa no es tan fácil ya que tenemos que llegar hasta la habitación del Dragón en un determinado tiempo.
A diferencia de otros juegos, el tiempo no es fijo. Nosotros podemos incrementarlo si recogemos unos paquetes estratégicamente ubicados en la base del dragón.
Por cada paquete tomado, el tiempo aumentará en veinte segundos. Justamente ahí está el "quid" de la cuestión. Los paquetes están, como dijimos, sobre distintos lugares. Para tomarlos debemos convertirnos en auto o en avión. Sólo cuando llegamos a la habitación del Dragón nos convertimos en un "robot revólver".
A través de él deberemos pegarle a Dragón quien, mientras tanto, nos tira un fuego que no es precisamente reconfortante. Claro que para matarlo no disponemos de todo el día. Lamentablemente tenemos límite de tiempo.
Si en una de esas le pegamos, pasaremos al siguiente nivel.

2-D MAZE



Rating Total: B
Creatividad: B
Profundidad: B
Valor en relación al precio: Se justifica
Computadora: Drean Commodore16
Editor: Melbourne House
Distriubuidor oficial: Graphic Games

Lo que nos espera en 2-D MAZE no es para nada fácil. Estamos dentro de un laberinto del cual debemos salir rápidamente.
A través de las teclas I y P podremos trasladarnos hacia la izquierda o hacia la derecha.
La barra espaciadora también nos servirá para salir de este oscuro y peligroso laberinto.

ZAPP

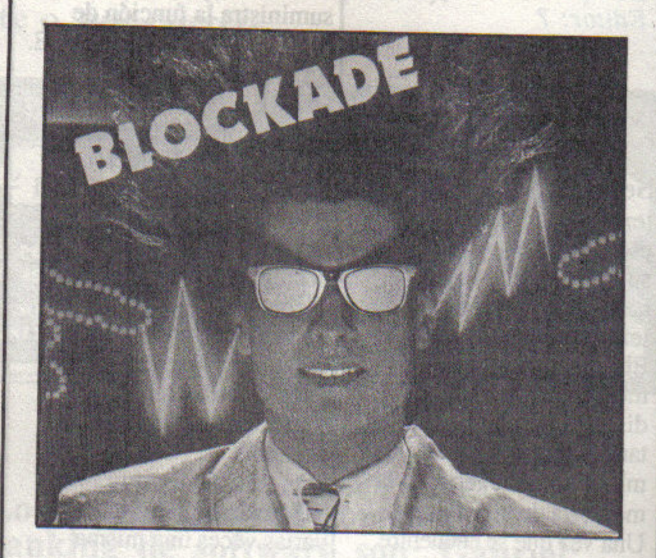


El juego termina cuando destruimos todas las bases de Suing. Por supuesto, no debemos descuidar todos los proyectiles que Suing nos dispara.
Las teclas que utilizamos para gobernar a nuestra nave son Q, Z, I, P y la barra espaciadora.
Ellas son las encargadas de subir, bajar, ir hacia la izquierda y derecha y disparar respectivamente.

Rating Total: B
Creatividad del juego: B
Profundidad: B
Valor en relación al precio: Se justifica
Computadora: Drean Commodore 16
Editor: Melbourne House
Distriubuidor oficial: Graphic Games

Luego de buscar y buscar hemos encontrado algunos programas disponibles para la Drean Commodore 16.
ZAPP es un juego en el que debemos disparar sobre las bases enemigas del emperador Suing.
Sólo tenemos dos disparos para contrarrestar al enemigo. No tenemos margen para fallar.

BLOCKADE



Rating Total: B
Creatividad del juego: B
Profundidad: B
Valor en relación al precio: Se justifica
Computadora: Drean Commodore 16
Editor: Melbourne House
Distriubuidor oficial: Graphic Games

Este es un juego donde participan dos jugadores. La idea consiste en construir una pared delante del adversario para que éste se la lleve por delante.
Cada jugador podrá controlar sus movimientos a través de las teclas Q y Z (jugador 1) y P e I (jugador 2).
Con ellas cada jugador mueve hacia izquierda o derecha.

HOMEWORD

Rating Total: B
Creatividad: B
Valor en relación al juego: Se justifica
Computadora: Drean Computadora 64/C
Editor: ?

Se trata de un procesador de texto que, a pesar de sus escasos bloques, le suministra al usuario herramientas de edición de texto sumamente potentes.
El programa, que como mencionamos viene en disco junto con su manual, tarda aproximadamente dos minutos en cargarse en la memoria de la máquina.
Una vez que se encuentre activado podemos comenzar a editar cualquier tipo de documentos, incluyendo cartas personalizadas.
Veamos algunas de las operaciones de edición. El borrado de texto se determina a través de las teclas de movimiento del cursor.
Con ellas se va seleccionando el texto a borrar, el cual se va "pintando" (en realidad, se va imprimiendo en video inverso) indicando así lo que se eliminará del documento.
La inserción de nuevos caracteres se realiza en forma automática, siendo el HOMEWORD el encargado de ir reacomodando el texto.
Por otra parte. el procesador permite trasladar subtextos de un lado a otro del documento en edición.
Por ejemplo, supongan que hemos escrito una oración la cual debe aparecer en otro momento.
Sólo bastará con marcarla (es decir "pintarla" con las teclas de movimiento del cursor) para que HOMEWORD las mueva (o inclusive copie) en otro lugar del documento.
Para la búsqueda de texto, se suministra la función de FIND y FIND&REPLACE.
En el primer caso, HOMEWORD la utiliza para encontrar una letra, palabra, frase u oración.
En muy pocos segundos, el procesador nos dirá la posición dentro del documento en donde se ha encontrado.
La segunda función nos permite buscar los mismos tipos de estructuras sintácticas y reemplazarlas por otras.
Esta función es muy solicitada cuando se cita varias veces una misma palabra en un texto.
Si esa palabra es, por ejemplo, "Señor", podemos abreviarla con "*". Luego, y antes de imprimirlo, le pedimos al procesador que busque el carácter "*" y lo reemplace por "Señor".
El formato del texto a imprimir se logra a través de LAYOUT. Con él, le indicamos a HOMEWORD cuáles serán los márgenes derechos e izquierdos del documento, espaciado entre línea y línea, y márgenes superior e inferior.
Además, es posible insertar encabezamientos y pies de página junto con el número de página.
En este caso es posible indicarle a HOMEWORD si empezamos con el número 1 o con otro cualquiera.
El tipo de letra que se utilizará en la impresión del texto puede elegirse de entre tres tipos distintos.
HOMEWORD nos pide que antes de sacar un documento por impresora le indiquemos qué tipo de impresora usaremos.
Podemos utilizar cualquier impresora Commodore como la MPS 801, 802, 803, 1000 o la nueva 1200.
En lo que respecta a las operaciones en disco, HOMEWORD dispone de comandos para guardar y recuperar documentos, eliminación y cambio de nombre de archivos, contenido del disco actual, formateo y reinicialización del disco, etcétera.

BAZOOKA BILL



Rating Total: B
Creatividad: B
Profundidad del juego: B
Valor en relación al juego: Se justifica
Computadora: Drean Commodore 64/C
Editor: Melbourne House
Distriubuidor oficial: Graphic Games

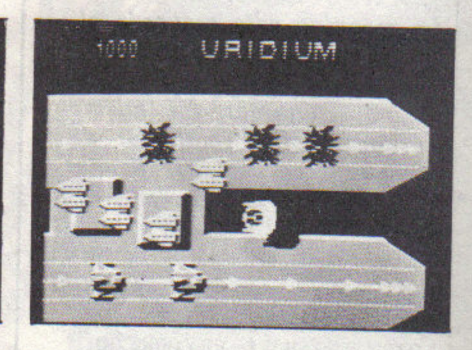
El general Mc. Arthur ha sido secuestrado por el ejercito revolucionario de las islas del Pacífico Sur, quien amenaza matarlo si los Estados Unidos invade sus islas.
El gobierno, sabiendo que no puede enviar un comando ya que los guerrilleros lo detectarían inmediatamente, eliminando así a Mc. Arthur, decide enviar a un veterano de Vietnam apodado "Bazooka" Bill.
Su misión consiste en rescatar al general norteamericano y regresarlo a los Estados Unidos.
A través de informes de inteligencia, se confirma que los rebeldes están equipados con pertrechos de combate sumamente sofisticados.
La tarea de "Bazooka" Bill no será nada fácil ya que el enemigo lo está esperando.
Utilizando el joystick tendremos que guiar a "Bazooka" a través de las islas.
Para trasladarse de isla en isla, Bill tendrá que encontrar las pistas de aterrizaje enemigas y robar un jet equipados con misiles y ametralladoras.

RANKING DE SOFTWARE

Por votación de los lectores se elegirán los cinco mejores programas de juegos o utilitarios creados para la computadora Drean Commodore 64C.
Para participar se debe enviar el cupón (o fotocopia) a nuestra Redacción: Paraná 720, 5º Piso, Cap. Fed. (1071), personalmente o por correo.

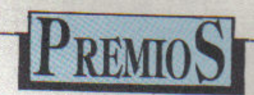






1º) Ghost'n Goblins
2º) 1942
3º) Uridium
4º) Ace
5º) Ace of ace

Los ganadores del sorteo del ranking de software son: Gabriel Belmonte, Ricardo Serres, Gustavo González, Javier Aquilo, Diego Correa, Mario Sanz, Horacio Parrondo, Gustavo Losada, Héctor Pasti, y Patricia Lisjak.



Entre los que envíen sus cupones se sortearán mensualmente:

Diez Software para Drean Commodore 64C

CUPON RANKING DE SOFTWARE DREAN COMMODORE

APELLIDO .... NOMBRE .... 1º ....
DIRECCION .... LOCALIDAD .... 2º ....
PROVINCIA .... C.P. .... T.E. .... 3º ....
OCUPACION .... EDAD .... 4º ....
COMPUTADORA .... 5º ....
QUE ES LO QUE MAS ME GUSTA DE LA REVISTA DEL USUARIO DE DREAN COMMODORE
....
QUE ES LO QUE MENOS ME GUSTA ....
....

Juntando programas

Ante todo deseo felicitarlos por la excelente labor que realizan con esta revista.
Pero como no hace demasiado tiempo que la compro, tengo algunas dudas respecto de uno de los programas publicados. Una de ellas se refiere a la manera de incorporar "Utiliprint" al programa "XXX" o viceversa.
La máquina solo recuerda el último programa grabado.
También quisiera saber si es posible combinar ese mismo utilitario con "Ampliador de caracteres" o con "Graficador de funciones".

Pablo Espina
Capital

Lo que necesitas es un comando especial que permita unir dos programas cualquiera en uno solo. Este comando suele llamarse MERGE y, lamentablemente, no se encuentra disponible en el Basic 2.0 de la Drean Commodore 64/C.
Si consigues el Simon's Basic, éste sí cuenta con este comando con el cual podrás unir varios programas en uno solo.

Club C-16

Antes que nada quiero que sepan que su revista es la mejor del mundo. Tengo dos pedidos:
Me gustaría formar un club de usuarios de Drean Commodore 16, para intercambiar ideas y datos.
Tengo muchísimos programas míos y copiados que deso compartir. De paso quisiera saber si puedo hacerles algún programa mío para que lo publiquen.

Guillermo Alfaro
Del Kaiser 1750, Martín Coronado, Tres de Febrero (1682), Bs. As.

Te felicitamos por tu iniciativa de formar un club de usuarios de Drean Commodore 16.

Continuamos con esta sección para que los lectores planteen sus consultas y sugerencias. Para eso deben escribir a Revista para usuarios de Drean Commodore, Paraná 720, 5to. Piso, (1017) Cap.

Desde ya te invitamos a ti y a todos aquellos lectores que quieran ver sus programas en la revista a que nos los envíen.

MPS 802

Antes que nada deseo felicitarlos por su excelente publicación. Deseo, además, hacerles la siguiente consulta.
Poseo una Drean Commodore 64 y una impresora MPS 802. Cuando utilizo la instrucción COPY del Simon Basic (en cartridge) se imprimen caracteres que no tienen nada que ver con lo expuesto en pantalla. Efectué la misma prueba utilizando para ello una impresora paralelo con su correspondiente interfase y obtuve resultados positivos. ¿Ustedes podrían decirme a qué se debe este defecto? ¿Existirá alguna forma de solucionar mi problema? Necesito que evacuen esta duda ya que necesito sacar gráficos por impresora. Desde ya les agradezco profundamente.

Juan Rodríguez
S. de Padua

Lo que sucede, estimado Juan, es que la MPS 802 no tiene modo gráfico, lo cual provoca que cuando ejecutas la instrucción COPY veas caracteres raros. La solución existe y, aunque te duela, consiste en que cambies la impresora por alguna que sí tenga modo gráfico como la MPS 801, 803 o la nueva 1000.
Drean también tiene una impresora gráfica, unidireccional, denominada DC770 que también te permitirá hacer gráficos.

MODEM CP/M

Antes que nada quiero felicitarlos por su excelente revista.
1) Una Drean Commodore 64 con modem de comunicaciones RS 232, ¿con cuántas computadoras puede comunicarme?
2) ¿Qué debo hacer para que mi Drean Commodore 64C trabaje en CP/M?

Rodolfo Maldonado
Buenos Aires

1) La comunicación entre computadoras es similar a la telefónica. Sólo podemos hablar con una persona a la vez, a menos que estemos conectados a un centro de recolección de llamadas el cual permite que varios usuarios hablen entre sí.
Con las computadoras pasa algo similar. Dando por descontado que para dialogar con otra computadora ésta debe hablar nuestro mismo lenguaje (BELL o CCITT), la comunicación se establece con otra computadora ubicada en el punto distante.
Y como en el caso anterior, existen bancos de datos que permiten que varios usuarios dialoguen (con sus computadoras) entre sí.
2) Hasta hace muy poco se vendía en los Estados Unidos un cartucho que contenía, entre otras cosas, un microprocesador Z80 que permitía correr CP/M 2.2 en la C-64.
Luego del lanzamineto de la C-128 que, como sabemos, trabaja en su tercer modo con CP/M +, ese cartucho perdió razón de ser.
En nuestro país ese cartucho no se comercializó debido a los altos costos de producción.
Como nos dijo uno de los fabricantes que iba a desarrollarlo, los costos de elaboración implicaban un alto precio de venta al público.
En resumidas cuentas, y un poco para aclararte tu segunda pregunta, si deseas trabajar con CP/M, sugerimos que compres una C-128.

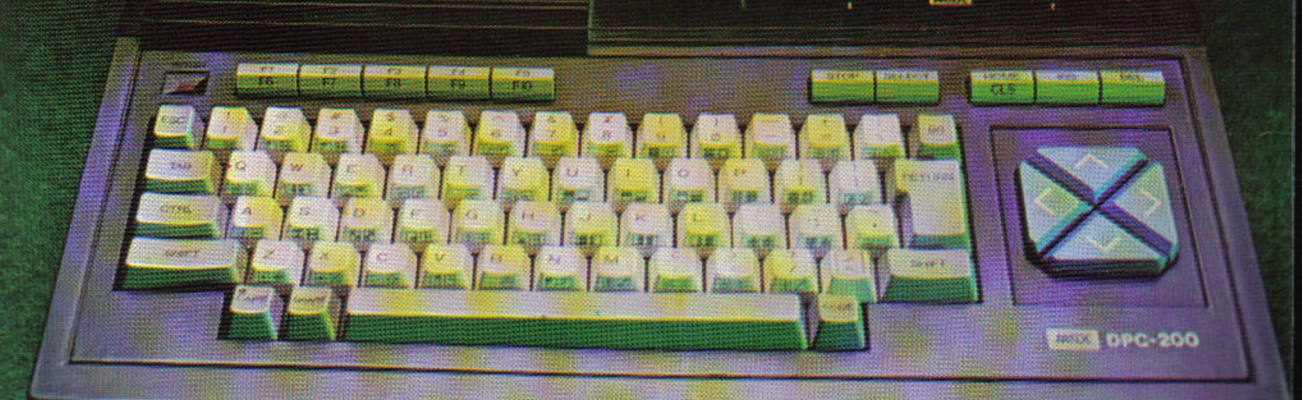
COMPUTACION K64 PARA TODOS

ABRIL 1987

NUEVOS

UTILITARIOS Y APLICACIONES

12 PROGRAMAS INEDITOS



NUMERO 25 A 3,90 REP. ARGENTINA

SHOCK INFORMATICO

HARD Y SOFT PARA SPECTRUM, CZ, TK y TS, DREAN COMMODORE, MSX, TI, y ATARI

Drean C= COMMODORE 64C

LA COMPUTADORA PERSONAL MAS VENDIDA DEL MUNDO!!

AHORA CON MAS PRESTACIONES!!

LA NUEVA DREAN COMMODORE 64 C INCORPORA EL PROGRAMA MAS NOVEDOSO
DE DIBUJO Y COMPOSICION DE TEXTOS.
ESCRIBE Y EDITA EN PANTALLA.
SELECCIONA 6 DIFERENTES TIPOS DE LETRAS EN 6 MEDIDAS DISTINTAS.
LE PERMITE DIBUJAR, PINTAR Y BORRAR EN PANTALLA.
DISEÑA CON 32 PATRONES.
PINTA EN 16 COLORES.







LA ULTIMA PALABRA EN TELECOMUNICACIONES

CON SU NUEVA DREAN COMMODORE 64 C,
PROVISTA DE UN MODEM, USTED PUEDE COMUNICARSE,
CON EL PAIS Y EL MUNDO MEDIANTE
EL 1º SERVICIO ARGENTINO
DE INFORMACIONES Y COMUNICACIONES
EN LINEA (DELPHI).
ADEMAS LE PERMITE INTERCAMBIAR
MENSAJES CON AMIGOS Y EL CLUB DE USUARIOS DREAN
COMMODORE, CON 25 FILIALES EN TODO EL PAIS
QUE LE BRINDARAN EL ASESORAMIENTO QUE USTED NECESITA.
ESTAS SON SOLO ALGUNAS COSAS
QUE USTED PUEDE HACER CON LA
NUEVA DREAN COMMODORE 64 C.

FABRICADO POR Drean SAN LUIS S.A.

A LA VANGUARDIA DE LA INFORMATICA EN ARGENTINA.